



# **Security — Configuration and Management Avaya Secure Router 2330/4134**

Release 10.3.5  
NN47263-600  
Issue 04.02  
August 2013

## Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

## Documentation disclaimer

"Documentation" means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya generally makes available to users of its products. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

## Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

## Warranty

Avaya provides a limited warranty on its hardware and Software ("Product(s)"). Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya's standard warranty language, as well as information regarding support for this Product while under warranty is available to Avaya customers and other parties through the Avaya Support website: <http://support.avaya.com>. Please note that if you acquired the Product(s) from an authorized Avaya reseller outside of the United States and Canada, the warranty is provided to you by said Avaya reseller and not by Avaya. "Software" means computer programs in object code, provided by Avaya or an Avaya Channel Partner, whether as stand-alone products or pre-installed on hardware products, and any upgrades, updates, bug fixes, or modified versions.

## Licenses

THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYA WEBSITE, [HTTP://SUPPORT.AVAYA.COM/LICENSEINFO](http://support.avaya.com/LICENSEINFO) ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AUTHORIZED AVAYA RESELLER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AUTHORIZED AVAYA RESELLER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA AUTHORIZED RESELLER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE ("AVAYA").

Avaya grants you a license within the scope of the license types described below, with the exception of Heritage Nortel Software, for which the scope of the license is detailed below. Where the order documentation does not expressly identify a license type, the applicable license will be a Designated System License. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation or other materials available to you. "Designated Processor" means a single stand-alone computing device. "Server" means a Designated Processor that hosts a software application to be accessed by multiple users.

## Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, or hardware provided by Avaya. All content on this site, the documentation and the Product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and/or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

## Third Party Components

"Third Party Components" mean certain software programs or portions thereof included in the Software that may contain software (including open source software) distributed under third party agreements ("Third Party Components"), which contain terms regarding the rights to use certain portions of the Software ("Third Party Terms"). Information regarding distributed Linux OS source code (for those Products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the Documentation or on Avaya's website at: <http://support.avaya.com/Copyright>. You agree to the Third Party Terms for any such Third Party Components.

## Note to Service Provider

The Product may use Third Party Components that have Third Party Terms that do not allow hosting and may need to be independently licensed for such purpose.

## Preventing Toll Fraud

"Toll Fraud" is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

## Avaya Toll Fraud intervention

If you suspect that you are being victimized by Toll Fraud and you need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: <http://support.avaya.com>. Suspected security vulnerabilities with Avaya products should be reported to Avaya by sending mail to: [securityalerts@avaya.com](mailto:securityalerts@avaya.com).

## Trademarks

The trademarks, logos and service marks ("Marks") displayed in this site, the Documentation and Product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation and Product(s) should be construed as granting, by implication, estoppel, or otherwise,

any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.

Avaya is a registered trademark of Avaya Inc.

All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

### **Downloading Documentation**

For the most current versions of Documentation, see the Avaya Support website: <http://support.avaya.com>.

### **Contact Avaya Support**

See the Avaya Support website: <http://support.avaya.com> for product notices and articles, or to report a problem with your Avaya product. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: <http://support.avaya.com>, scroll to the bottom of the page, and select Contact Avaya Support.



## Contents

<b>Chapter 1: Introduction.....</b>	<b>17</b>
Purpose.....	17
Related Resources.....	17
Documentation.....	17
Training.....	17
Avaya Mentor videos.....	17
Support.....	18
<b>Chapter 2: New in this release.....</b>	<b>19</b>
Other changes.....	19
DOS protection option tcp-seq-except-bgp-self-port.....	19
Secure FTP (SFTP) client.....	19
Performance tuning for firewall on GRE/IP-IP tunnels.....	19
Support for multiple Avaya VPN clients behind NAT.....	20
<b>Chapter 3: Firewall and NAT Fundamentals.....</b>	<b>21</b>
Firewall Overview.....	21
Stateful inspection elements.....	22
Virtual firewall zones.....	22
Transit policies on trusted zones only.....	23
No transit policies on internet untrusted zone.....	23
Default firewall.....	24
Simple firewall failover using RST.....	24
Three-legged firewall.....	26
Firewall network protection features.....	26
Application and URL filtering.....	27
Policy-based controls.....	27
Logging and statistics.....	28
ALG Overview.....	28
Supported ALGs.....	28
NAT Overview.....	30
Static NAT.....	31
Dynamic NAT.....	32
PAT.....	32
NAT failover for firewalls.....	32
Scalability.....	33
Call server interoperability.....	33
NAT traversal strategies for VoIP.....	34
NAT ACL enhancements.....	34
NAT ACL Packet Processing.....	34
Cone NAT.....	35
SIP ALG.....	35
Line-side SIP traffic translation.....	36
Trunk-side SIP traffic translation.....	37
NAT Hairpinning.....	38
Scenarios.....	38

Firewall only.....	39
Forward NAT.....	39
Forward NAT with Hairpinning.....	40
Reverse NAT.....	40
Servers and scenarios supported in Secure Router 2330/4134.....	40
Multimedia Communication Server (MCS).....	40
Live Communication Server (LCS).....	41
Office Communication Server (OCS).....	41
Sylantro and Broadsoft servers.....	41
Avaya Communication Server 1000.....	41
SIP ALG support for Media Gateway and SSM in Secure Router.....	42
Firewall only with Media Gateway.....	43
Forward NAT With Media Gateway.....	43
Forward NAT with SSM.....	43
Standards compliance.....	43
<b>Chapter 4: Packet filter fundamentals.....</b>	<b>45</b>
Packet filters on specific Secure Router chassis.....	45
Packet filters on the Secure Router 4134.....	45
Packet filters on the Secure Router 2330.....	47
Packet filter relationship to other Secure Router 2330/4134 features.....	48
Packet filter relationship to firewall and VPN.....	48
Packet filter relationship to PCAP.....	50
Packet filter relationship to QoS.....	50
Available packet filters.....	50
IPv4 packet filters.....	51
IPv6 packet filters.....	51
MAC packet filters.....	52
Packet filters for management services.....	52
Packet filter logging.....	53
Packet filter scalability and capacities.....	54
Secure Router 4134.....	54
Secure Router 2330.....	54
Packet filter configuration considerations.....	55
Packet filter limitations and restrictions.....	55
Packet filter troubleshooting.....	56
<b>Chapter 5: IPsec VPN fundamentals.....</b>	<b>57</b>
Site-to-Site VPN.....	57
Tunnel failover using static weighted tunnels.....	59
Tunnel failover using round robin DNS.....	60
IPSec VPN bypass policy.....	62
IPSec nailed up tunnel.....	63
Remote access VPN.....	63
Remote access VPN with L2TP server.....	64
Supported IPsec security protocols.....	65
IPsec modes.....	66
Shared key negotiation with IKE.....	66
IKE modes.....	66

Key usage extension checking.....	68
Peer authentication methods for IKE.....	69
Client configuration and user authentication for remote access VPN.....	69
Digital Certificates in IKE.....	70
Internet X.509 PKI certificate and CRL profile.....	71
X.509 digital certificate compliance with RFC 2253.....	71
Certificate validation.....	73
Certificate enrollment using SCEP client.....	74
Manual certificate enrollment.....	74
RSA certificate key size.....	75
Dead peer detection.....	76
Nat Traversal support.....	76
Larger DH groups for branch office tunnels.....	77
Multiple IKE proposals.....	77
Multiple IPsec proposals.....	78
Multiple networks in a single IPsec policy.....	79
Identifying traffic to be encrypted with VPN.....	80
Prioritizing IPsec policies.....	81
Firewall considerations for trusted and untrusted VPN interfaces.....	81
Routing considerations for VPN (and firewall).....	82
Perfect forward secrecy.....	82
Security Policy Database.....	82
PMTU support.....	83
TCP MSS Clamping.....	83
Firewall considerations with VPN.....	84
IPsec VPN support without firewall.....	84
QoS over VPN.....	85
Crypto QoS (CBQ) for IPsec VPN.....	85
Logging and Statistics.....	87
Standards compliance.....	87
<b>Chapter 6: Avaya VPN client fundamentals.....</b>	<b>89</b>
Overview of Avaya VPN client operation.....	89
Remote management using Avaya VPN client.....	90
Supported Avaya VPN features.....	91
Client version recognition.....	91
Authentication.....	92
Proprietary IKE Keepalive.....	95
Standard Client Configuration Push.....	96
Extended Client Configuration Push.....	96
Extra IKE status codes.....	103
Avaya VPN client Interoperability.....	104
Phase 1 proposals.....	105
Phase 2 proposals.....	105
Rekey Behavior.....	106
Mandatory client configuration parameters.....	106
Banner Text.....	106
Avaya VPN client interactions with routing.....	107

<b>Chapter 7: GRE and IPIP tunneling fundamentals.....</b>	<b>109</b>
GRE and IPIP tunneling for IPv4.....	109
IPIP.....	110
GRE.....	110
Tunnel protection.....	110
VLAN over GRE.....	110
PCAP over GRE.....	111
IPv6 over IPv4 tunneling.....	111
IPv6 over manually-configured IPv4 tunnels.....	111
IPv6 over IPv4 GRE tunnels.....	111
Auto 6to4 tunneling.....	112
Standards compliance.....	112
<b>Chapter 8: PPPoE client fundamentals.....</b>	<b>113</b>
Standards compliance.....	114
<b>Chapter 9: Authentication, Authorization, and Accounting fundamentals.....</b>	<b>115</b>
Authentication.....	115
PAP authentication.....	115
CHAP authentication.....	115
RADIUS.....	116
TACACS.....	116
EAP IEEE 802.1X.....	118
RSA SecureID.....	119
Standards compliance.....	122
Authorization.....	122
Accounting.....	123
TACACS accounting.....	123
<b>Chapter 10: SSH2 fundamentals.....</b>	<b>125</b>
SSH2 features.....	125
SSH ciphers.....	126
SSH MAC algorithms.....	126
SSH compression.....	127
SSH key exchange methods.....	127
SSH public key algorithms.....	127
SSH user authentication methods.....	128
SSH public key file formats.....	128
Standards compliance.....	128
<b>Chapter 11: Firewall and NAT configuration.....</b>	<b>131</b>
Configuring global properties.....	131
Configuring global ALGs.....	131
Configuring the DNS ALG.....	132
Configuring global bypass trusted.....	133
Configuring global DOS protection.....	134
Configuring global NAT hairpinning.....	136
Configuring peer-to-peer RTP media.....	137
Configuring self firewall policy with NAT.....	137
Configuring global IP reassembly.....	138
Configuring global logging.....	141



Configuring global maximum connection limits for the firewall.....	142
Configuring NAT ACL.....	143
Configuring NAT failover.....	144
Configuring proxy NAT.....	145
Configuring RFC 3947 NAT traversal acceptance.....	145
Configuring global timeout.....	146
Configuring global URL key filters.....	147
Configuring port trigger records.....	148
Configuring policy-specific properties.....	149
Configuring firewall objects.....	149
Configuring connection reservations.....	151
Configuring reset of invalid ACK packets.....	151
Configuring stealth mode.....	152
Configuring firewall policies.....	152
Applying an object to a policy.....	154
Configuring bandwidth for the policy.....	154
Configuring the maximum connections for the policy within a configured timeframe.....	155
Configuring the maximum connections for the policy.....	156
Configuring policing for the policy.....	156
Enabling the policy.....	157
Adding interfaces to the firewall zone.....	157
Displaying firewall information.....	158
Clearing firewall connections.....	159
Clearing firewall statistics.....	159
<b>Chapter 12: Packet filter configuration.....</b>	<b>161</b>
Configuring IPv4 packet filters.....	161
Configuring IPv6 packet filters.....	164
Configuring MAC packet filters.....	167
Applying a packet filter to an interface.....	169
Applying a management services packet filter.....	169
Deleting rules from packet filters.....	170
Deleting a packet filter.....	171
Clearing packet filter counters.....	171
Clearing packet filter statistics.....	172
Displaying packet filters.....	172
Displaying packet filters applied to an interface.....	172
Displaying global management service packet filter information.....	173
Displaying global IPv4 management service packet filter statistics.....	173
Displaying global IPv6 management service packet filter statistics.....	173
<b>Chapter 13: IPsec VPN configuration.....</b>	<b>175</b>
Configuring IKE for site-to-site VPN.....	176
Creating an IKE policy.....	176
Configuring the local address for IKE negotiations.....	176
Configuring the IKE policy local ID.....	177
Configuring the IKE policy remote ID.....	177
Configuring the IKE mode.....	178
Configuring the IKE exchange type.....	179

Configuring the pre-shared key for IKE.....	180
Configuring key usage extension checking.....	180
Enabling or disabling PFS.....	181
Configuring IKE proposal.....	181
Configuring OCSP for the IKE policy.....	185
Configuring IPsec for site-to-site VPN.....	186
Creating an IPsec policy.....	186
Configuring ABOT tunneling.....	187
Configuring anti-replay.....	188
Enabling or disabling the IPsec policy entry.....	189
Configuring an IPsec VPN bypass policy.....	189
Enabling IPsec nailed up tunnel.....	191
Specifying the IP stream on which to apply IPsec.....	192
Configuring DH prime modulus group for PFS.....	194
Assigning multiple networks to a single IPsec policy.....	194
Configuring IPsec proposal.....	197
Configuring static weighted tunnels for tunnel failover.....	201
Configuration example — Tunnel failover using round robin DNS.....	202
Configuration example — Tunnel failover using static weighted tunnels.....	203
Configuring remote access IKE policies.....	210
Creating an IKE policy for remote access VPN.....	210
Configuring an IKE proposal for remote access VPN.....	217
Configuring remote access IPsec policies.....	222
Creating an IPsec policy for remote access VPN.....	222
Specifying the IP stream on which to apply IPsec for remote access VPN.....	222
Configuring DH prime modulus group for PFS.....	224
Configuring IPsec proposal template for remote access VPN.....	224
Enabling the dynamic IPsec policy.....	228
Configuring L2TP server for L2TP remote access.....	229
Creating the L2TP remote access interface.....	229
Configuring IP address for the L2TP access interface.....	229
Configuring IPsec protection for the L2TP access interface.....	229
Configuring client parameters for L2TP remote access.....	230
Configuring user parameters for L2TP remote access.....	231
Shutting down the L2TP access interface.....	231
Configuring dead peer detection keepalive.....	232
Enabling dead peer detection.....	232
Configuring the DPD mode.....	232
Configuring the DPD keepalive retry interval.....	233
Configuring the DPD keepalive transmit interval.....	233
Configuring PMTU.....	234
Configuring DF bit.....	234
Configuring the MTU threshold value.....	235
Configuring processing of unsecured ICMP messages.....	235
Configuring TCP MSS on an Ethernet interface.....	235
Configuring CA trustpoint.....	236
Configuring the certificate enrollment method.....	236

Configuring parameters for the certificate request.....	237
Configuring certificate password.....	240
Authenticating the CA and importing a CA certificate.....	240
Generating a certificate request for enrollment.....	241
Manually importing a self certificate.....	241
Automatically importing a self certificate through SCEP.....	242
Manually importing an OCSP Responder certificate.....	243
Configuring LDAP parameters.....	243
Requesting a CRL from the CA.....	244
Configuring OCSP.....	245
Configuring IPsec VPN Support without Firewall.....	245
Displaying IPsec VPN configurations.....	246
Displaying certificates.....	246
Displaying CRL.....	246
Displaying trustpoint.....	246
Displaying IKE policies.....	246
Displaying IKE SA.....	246
Displaying IPsec policies.....	247
Displaying IPsec SA.....	247
Displaying remote access IKE policies.....	247
Displaying remote access IPsec policies.....	247
Displaying remote access VPN clients.....	248
Displaying status of interfaces as trusted or untrusted.....	248
Displaying dead peer detection configuration.....	248
Displaying PMTU information.....	248
Displaying IPsec statistics.....	248
Displaying L2TP server configuration.....	248
Clearing IPsec configurations.....	249
Deleting certificates.....	249
Deleting CRL.....	249
Deleting CA private key.....	249
Clearing IKE SA information.....	249
Clearing IPsec SA information.....	249
Clearing IPsec statistics.....	250
<b>Chapter 14: Avaya VPN client configuration.....</b>	<b>251</b>
Configuring IKE policy parameters for Avaya VPN client.....	252
Creating an IKE policy for Avaya VPN client.....	252
Configuring the IKE policy local address for Avaya VPN client.....	252
Configuring the IKE policy remote ID for Avaya VPN client.....	253
Configuring IKE mode for Avaya VPN client.....	254
Configuring mode configuration client parameters for Avaya VPN client.....	255
Configuring an address pool for mode configuration.....	255
Configuring a private side address.....	256
Configuring DNS server address for mode configuration.....	256
Configuring a WINS server address for mode configuration.....	257
Configuring split tunnel parameters.....	258
Configuring the client domain name.....	259

Configuring whether the client can store username and password.....	260
Configuring the client screen saver wait time.....	261
Configuring the client banner text.....	261
Enabling or disabling the client banner.....	262
Configuring failover for the Secure Router.....	263
Configuring keepalive behavior.....	264
Configuring NAT keepalives.....	265
Configuring IKE proposal parameters for Avaya VPN client.....	266
Configuring an IKE proposal for Avaya VPN client.....	266
Configuring authentication method for IKE proposal for Avaya VPN client.....	266
Configuring DH group for IKE proposal for Avaya VPN client.....	267
Configuring encryption algorithm for IKE proposal for Avaya VPN client.....	268
Configuring IKE hash algorithm for Avaya VPN client.....	269
Configuring IPsec proposal parameters for Avaya VPN client.....	270
Creating an IPsec policy for Avaya VPN client.....	270
Configuring encryption algorithm for IPsec proposal for Avaya VPN client.....	270
Configuring the hash algorithm for IPsec proposal for Avaya VPN client.....	271
Configuring lifetime for IPsec proposal for Avaya VPN client.....	272
Displaying Avaya VPN client configuration.....	273
Displaying IKE policies.....	273
Displaying IPsec policies.....	273
Displaying Avaya VPN clients.....	273
<b>Chapter 15: GRE and IPIP tunnel configuration.....</b>	<b>275</b>
Configuring a tunnel.....	275
Creating a tunnel.....	275
Configuring tunnel encapsulation mode.....	275
Configuring an IP address for the tunnel.....	276
Configuring tunnel source.....	277
Configuring tunnel destination.....	277
Configuring TCP MSS on a GRE/IPIP tunnel interface.....	278
Configuring GRE tunnel parameters.....	279
Configuring keepalive for GRE tunnels.....	279
Configuring checksum for GRE tunnels.....	279
Configuring tunnel key for GRE tunnels.....	280
Configuring tunnel sequencing.....	280
Configuring tunnel parameters.....	281
Configuring path MTU discovery for tunnel packets.....	281
Configuring the tunnel as an untrusted interface for IPsec protection.....	281
Configuring tunnel protection with IPsec.....	282
Configuring tunnel ToS.....	282
Configuring tunnel TTL.....	283
Shutting down a tunnel.....	283
Configuring VLAN parameters for the tunnel.....	284
Configuring PCAP over GRE.....	284
Displaying tunnel information.....	284
Clearing tunnel counters.....	285
<b>Chapter 16: PPPoE client configuration.....</b>	<b>287</b>

Creating a PPPoE interface.....	287
Configuring IP address for PPPoE interface.....	287
Configuring PPPoE tunneling protocol.....	288
Configuring PPPoE Ethernet interface.....	288
Configuring PPP authentication method and parameters.....	289
Configuring PPPoE access concentrator.....	289
Configuring PPP keepalive.....	290
Displaying PPPoE client information.....	290
<b>Chapter 17: Authentication, Authorization, and Accounting configuration.....</b>	<b>291</b>
Enabling AAA.....	291
Configuring AAA authentication.....	291
Configuring AAA authentication login.....	291
Configuring AAA authentication protocol.....	292
Applying AAA authentication to an interface.....	293
Configuring AAA authorization.....	293
Configuring AAA authorization.....	293
Applying AAA authorization to an interface.....	294
Configuring AAA accounting.....	294
Configuring AAA accounting.....	294
Configuring AAA accounting update.....	295
Applying AAA accounting to an interface.....	295
Configuring the AAA source address.....	296
Configuring RADIUS primary and secondary servers.....	296
Configuring RADIUS server port for accounting.....	296
Configuring RADIUS server port for authentication.....	297
Configuring the RADIUS server IP address.....	297
Configuring RADIUS client retries.....	298
Configure RADIUS shared secret key.....	298
Configure RADIUS timeout.....	299
Configuring RADIUS client source address.....	299
Configuring TACACS accounting.....	300
Configuring TACACS+ primary or secondary server IP address.....	301
Configuring TACACS+ retries.....	301
Configuring TACACS+ server port.....	301
Configuring TACACS+ shared encryption key.....	302
Configuring TACACS+ timeout.....	302
Configuring 802.1x.....	303
Configuring 802.1x on an Ethernet interface.....	303
Enable 802.1x on the interface.....	303
Configuring the maximum failed requests.....	304
Configuring port control.....	304
Configuring quiet period.....	305
Enabling reauthentication.....	305
Configuring reauthorization period.....	306
Configuring authentication server response timeout.....	306
Configuring supplicant response timeout.....	307
Displaying AAA information.....	307

Displaying AAA accounting information.....	307
Displaying AAA authentication information.....	307
Displaying AAA authorization information.....	307
Displaying AAA interface information.....	308
Displaying AAA status.....	308
Displaying RADIUS information.....	308
Displaying TACACS+ information.....	308
Displaying 802.1x information.....	308
Clearing 802.1x statistics.....	308
<b>Chapter 18: SSH2 configuration.....</b>	<b>311</b>
Configuring SSH2 server keys.....	311
Generating SSH2 server keys.....	311
Encrypting a private key file.....	312
Changing the passphrase used for encryption.....	312
Converting public key files to SSH format.....	313
Generating a public key digest of a key file.....	313
Configuring SSH2 server parameters.....	314
Configuring SSH2 authentication.....	314
Configuring SSH2 authentication retries.....	314
Configuring SSH encryption algorithms.....	315
Configuring SSH compression.....	316
Enabling and disabling SSH server.....	316
Specifying host key file for the SSH server.....	316
Enabling and disabling log events.....	317
Configuring MAC algorithms.....	317
Configuring SSH listen port.....	318
Restoring default SSH parameter values.....	318
Enabling and disabling SSH SFTP server.....	319
Configuring SSH session timeout.....	319
Configuring an SFTP client.....	320
Displaying SSH server configuration.....	321
Displaying SSH server sessions.....	321
Clearing SSH sessions.....	321
<b>Chapter 19: Configuration examples.....</b>	<b>323</b>
Configuring an IPv4 packet filter.....	323
Configuring an IPv6 packet filter.....	324
Configuring a MAC packet filter.....	326
Configuring a default firewall policy.....	327
Configuring a simple firewall policy with DMZ.....	328
Configuring a simple PAT policy.....	330
Configuring a PAT policy with an inbound forwarding policy.....	331
Configuring SIP ALG line-side.....	332
Configuring SIP ALG trunk-side.....	333
Configuring a Site-to-site IPsec VPN.....	334
Configuring a remote access IPsec VPN.....	336
Configuring a trust point for PKI.....	338
Configuring a remote access VPN with L2TP server.....	338

Configuring an IPv4 tunnel.....	338
VLAN over a GRE tunnel configuration example.....	341
Configuring an auto 6to4 tunnel.....	342
Configuring the firewall for NAT and IPsec tunnels.....	344
Configuring a PPPoE client.....	346
Secure Router 2330/4134 configuration for dynamic route exchange over IPsec tunnel interoperability with VPN Router.....	348
IP phone configuration for Secure Router 2330/4134.....	350
Secure Router 2330/4134 Interface, IPsec and Firewall configuration examples.....	350
Configuring an IPSec Tunnel between Avaya 96xx Series IP Phones and the Avaya Secure Router 2330/4134.....	354
<b>Chapter 20: Default settings.....</b>	<b>359</b>
Firewall command defaults.....	359
Packet filter defaults.....	362
IPSec VPN default settings.....	362
PPPoE default settings.....	366
GRE and IPIP tunnel default settings.....	367





# Chapter 1: Introduction

---

## Purpose

This document describes the operation and configuration of the security features on the Avaya Secure Router 2330/4134.

---

## Related Resources

---

### Documentation

See the *Avaya Secure Router 2330/4134 Documentation Roadmap*, NN47263-103, for a list of the documentation for this product.

---

### Training

Ongoing product training is available. For more information or to register, you can access the Web site at <http://avaya-learning.com>.

---

### Avaya Mentor videos

Avaya Mentor is an Avaya-run channel on YouTube that includes technical content on how to install, configure, and troubleshoot Avaya products.

Go to <http://www.youtube.com/AvayaMentor> and perform one of the following actions:

- Enter a key word or key words in the Search Channel to search for a specific product or topic.
- Scroll down Playlists, and click the name of a topic to see the available list of videos posted on the site.

---

## Support

Visit the Avaya Support website at <http://support.avaya.com> for the most up-to-date documentation, product notices, and knowledge articles. You can also search for release notes, downloads, and resolutions to issues. Use the online service request system to create a service request. Chat with live agents to get answers to questions, or request an agent to connect you to a support team if an issue requires additional expertise.

# Chapter 2: New in this release

There are no new features for *Avaya Secure Router 2330/4134 Security — Configuration and Management*, (NN47263-600) for Release 10.3.5.

## Important:

In this document, the term Secure Router 2330/4134 is used interchangeably to refer to the Secure Router 2330 and the Secure Router 4134. Content specific to either the Secure Router 2330 or the Secure Router 4134 is identified as such.

---

## Other changes

The following sections include security specific feature information that was previously documented in the Secure Router 2330/4134 release notes, and has been moved to *Avaya Secure Router 2330/4134 Security — Configuration and Management* (NN47263-600) with Release 10.3.5:

---

### DOS protection option `tcp-seq-except-bgp-self-port`

Release 10.3.5 supports the DOS protection option `tcp-seq-except-bgp-self-port`, which is used to allow the BGP destination port to not resequence. For information, see [Configuring global DOS protection](#) on page 134.

---

### Secure FTP (SFTP) client

Release 10.3.5 supports Secure FTP (SFTP) clients. For information on this feature, see [Configuring SFTP client](#) on page 320.

---

### Performance tuning for firewall on GRE/IP-IP tunnels

Release 10.3.5 supports transit traffic passing through an IP-IP tunnel.

For information, see [Transit policies on trusted zones only](#) on page 23.

---

## Support for multiple Avaya VPN clients behind NAT

Previously, when multiple Avaya VPN clients were behind NAT, they each needed a separate user profile if there was concurrent access. In Release 10.3.5, all clients can use the same user profile as long as the `keepalive` and `nat keepalive` parameters are both enabled. For information, see [Supported Extended Client Configuration Push features](#) on page 96.

# Chapter 3: Firewall and NAT Fundamentals

This chapter contains information to help you understand Firewall and Network Address Translation (NAT) Fundamentals.

---

## Firewall Overview

Firewalls perform a critical role in perimeter security, protecting network resources by determining who can access what on the network. To provide this protection, a firewall is placed at the gateway (or node) at which a secure network and an insecure network intersect; typically where the internal network for an organization intersects the Internet.

The firewall is a set of programs restricting incoming and outgoing traffic between the Internet and the internal network according to user-specified parameters. As a general rule, all network traffic, inbound and outbound, flows through the firewall. The firewall screens all incoming traffic and blocks that which does not meet the restrictions of the security policy.

There are three different types of firewall technologies available, namely, the packet filter, application proxy, and stateful inspection firewall.

The central idea of a packet filter is that a set of rules are defined to monitor inbound connections at the network layer level. As long as the inbound packet conforms to the defined rules, the packet is allowed to pass. While packet filters represent the fastest approach, with this solution, opened ports remain open indefinitely. Also, for certain applications to function properly, a wide range of ports must be opened. These two issues create vulnerabilities within the packet filter solution.

The application proxy terminates the incoming connection from the untrusted side, examines the packet in its entirety (up to the application layer) and chooses to forward it further or not. While this is the safest method, it is slower and highly application sensitive.

The approach that the Avaya Secure Router 2330/4134 implements is a stateful inspection firewall. The stateful inspection firewall relies on building network connections and monitoring their state to make a decision on admitting an inbound packet. All traffic passing through the stateful inspection firewall is analyzed against the state of the network connections in order to determine whether it is allowed to pass through. In a typical setup, only outbound rules are defined to permit or deny certain types of traffic. When allowing a packet to go from the trusted to untrusted network based on a rule match, the stateful firewall creates a network connection, which is uniquely identified by certain elements of the packet. Based on the application type (and the corresponding protocol), the appropriate inbound policy is dynamically created. When a return packet is received, the packet is allowed as long as the state of the network connection allows reception of this packet.

The inbound policy is a temporary policy that expires upon the expiry of the network connection. Since the inbound policy does not keep ports open indefinitely, network vulnerability is drastically reduced in comparison to a packet filter.

---

## Stateful inspection elements

Typically, a stateful firewall connection is identified by the following five basic elements:

- Source Address
- Destination Address
- Source Port
- Destination Port
- Protocol

Additional elements can also be included in the firewall connection for some special protocols.

---

## Virtual firewall zones

Most firewalls apply rules to single interfaces, but with the Secure Router 2330/4134, this is not the case. Instead, the Secure Router 2330/4134 firewall places interfaces into rule sets called virtual firewalls or zones. The advantage of configuring common rule sets is that you can perform the most complex task (editing rules) once, and apply this configuration across multiple interfaces. You no longer need to repeat policy definitions on multiple interfaces. Once a policy is defined for a zone, you can place any number of interfaces into that zone. As a result, the Secure Router 2330/4134 can accommodate complex policy configurations with less duplication of rule entry.

The Secure Router 2330/4134 provides a default zone, the internet, that is the only available untrusted zone. You must therefore add all untrusted interfaces to the internet zone. You cannot create a second untrusted zone.

The Secure Router 2330/4134 does not trust inbound connections on interfaces that are in the untrusted internet zone. These connections are blocked by default. Only interfaces from within a trusted zone are trusted to start new connections.

Outside of the internet zone, all other zones on the router are trusted zones. No traffic is allowed into a trusted zone unless a session is initiated from within that zone.

By default, all outbound connections from the trusted zone are allowed and all inbound connections are denied.

There is one default trusted zone, named corp. You can create additional zones as required to meet the needs of your network. All additional zones that you create are trusted zones.

---

## Transit policies on trusted zones only

Transit policies allow traffic to flow through the firewall. In order for traffic from the untrusted zone to flow back into the trusted zone, the session must be initiated from within the trusted zone. To configure rules for this traffic, you must configure transit policies on the trusted zones.

Each trusted zone has a database of incoming and outgoing policies. Outgoing policies are applied to traffic flowing from the trusted interfaces to the Internet. Incoming policies are applied to traffic flowing from the Internet to the trusted interface in the Virtual Firewall.

In addition, to allow traffic between trusted zones, you must configure the trusted zones to allow the transit traffic to pass. If desired, you can globally disable firewall processing between the trusted zones.

### Note:

Starting with Release 10.3, Avaya redesigned the Secure Router 2330/4134 optimized forwarding path to include the transit traffic passing through an IP-IP tunnel. Typically, tunnel interfaces are declared 'untrusted'. However, it is possible to declare a tunnel interface 'trusted' if that better matches the network topology. If a tunnel interface is declared 'trusted', the global firewall configuration option `bypass-trusted` must not be enabled.

---

## No transit policies on internet untrusted zone

Policies in the internet zone do not deal with transit traffic.

The primary use for the internet zone is to mark your untrusted interfaces.

The secondary use for the internet zone is to allow connections to the Secure Router 2330/4134 itself from the Internet (for example, to allow for a Telnet connection to the router from the Internet). Policies in the internet zone deal only with traffic destined to or sourced from an interface in the internet zone (or 'internet-self' traffic). These "self" policies are the only type of policy supported in the untrusted zone.

You cannot configure the internet zone with policies for traffic passing through the router. If you want to allow connections from the internet to travel into the trusted side, then you must configure an inbound policy in a trusted zone (like "corp").

The internet zone is not involved in governing transit traffic.

---

## Default firewall

The following figure shows the default firewall configuration: a corp trusted zone with all outbound connections allowed and an internet untrusted zone with all incoming connections blocked. In addition, all ALGs are enabled.

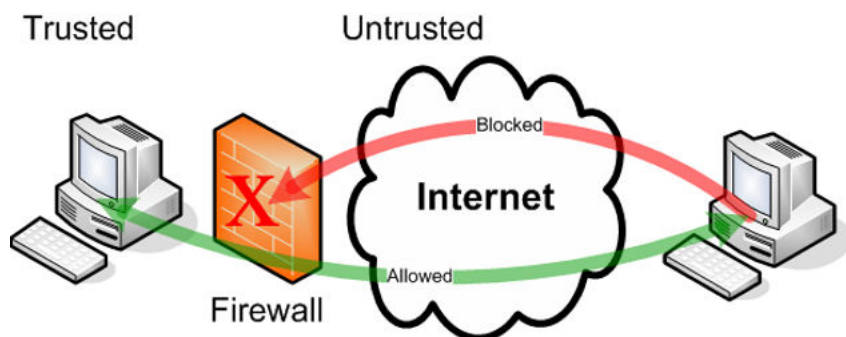


Figure 1: Default firewall

---

## Simple firewall failover using RST

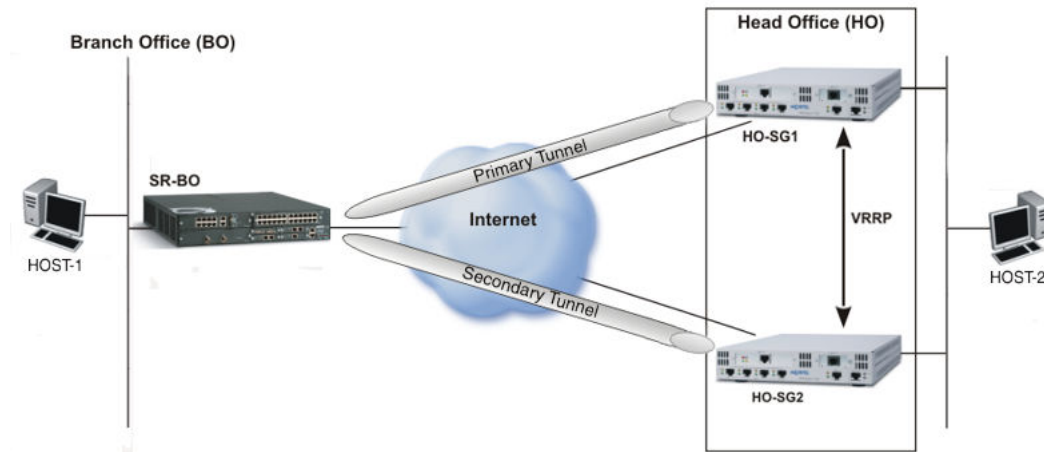
The simple firewall failover using RST feature is designed to reduce recovery time for a scenario where a network with two head office routers, running Virtual Router Redundancy Protocol (VRRP), are connected to a branch office Secure Router using primary and backup IPSec tunnels, and the primary tunnel connection fails.

This scenario applies for the network shown in the **Simple firewall failover using RST** network diagram, where the primary head office router (HO-SG1) and the secondary head office router (HO-SG2) have VRRP running between them.

The branch office Secure Router (SR-BO) is connected to HO-SG1 using a primary IPSec tunnel and connected to HO-SG2 using a secondary (backup) IPSec tunnel.

Firewall is enabled for the network.





**Figure 2: Simple firewall failover using RST**

If the primary tunnel connection fails, VRRP switches to HO-SG2 and traffic from SR-BO is directed over the backup tunnel.

Under normal operation, an enabled firewall considers any TCP data packets that are not preceded by TCP control packets to be a possible malicious attack, and the firewall discards the TCP data packets. Therefore, for the scenario shown in the **Simple firewall failover using RST** network diagram, with the absence of TCP control packets, the firewall discards TCP data packets, which disrupts traffic flow between SR-BO and HO-SG2.

With the simple firewall failover using RST feature, the time period for the network to recover from this firewall behavior is reduced.

Although TCP data packets are initially discarded by the firewall, with simple firewall failover using RST, the firewall sends a TCP:RST packet to Host-1 and Host-2, instructing both hosts to reset the TCP connection. However, because there is a brief delay before the secondary tunnel between HO-SG2 and SR-BO is fully established, Host-1 cannot initially receive the TCP:RST packet.

When the secondary tunnel between HO-SG2 and SR-BO is fully established, Host-1 attempts to send TCP data through the tunnel. The firewall again discards this traffic and sends a TCP:RST packet to Host-1 and Host-2, instructing both hosts to reset the TCP connection. Because Host-2 already received a TCP:RST packet, the host discards this second packet.

After the connection is reset, TCP control packets traverse the firewall, TCP data packets are accepted, and network traffic flow is restored over the secondary tunnel.

You must disable stealth-mode for simple firewall failover using RST to function.

## Three-legged firewall

A typical three-legged firewall configuration consists of the following three virtual firewalls:

- Corp – An organization's private trusted network
- Dmz – An organization hosting its web/ ftp/ mail server to public
- Internet – The public untrusted network

In this configuration, untrusted SSH and IKE connections to the router itself are allowed. Trusted and untrusted HTTP connections to a DMZ web server are allowed. And the default, corp trusted zone with all outbound connections are allowed.

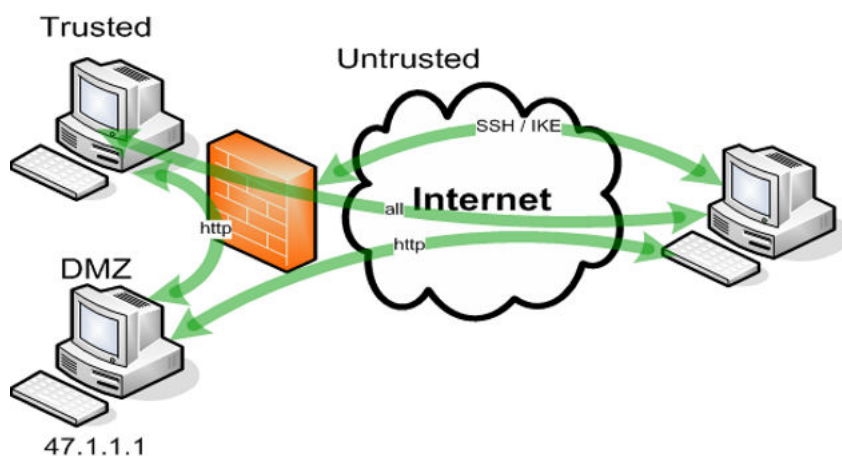


Figure 3: Zones

## Firewall network protection features

The Secure Router 2330/4134 firewall supports the following features to protect the network:

- Denial of Service protection

The Secure Router 2330/4134 firewall protects trusted networks from over 60 known malicious attacks, including but not limited to the following:

- Denial of Service (DOS) attacks that include: SYN-flood, Smurf, LAND, Ping of Death, Jolt, Jolt2, IP spoofing, Sequence number prediction, IP and transport protocol header integrity attacks, and WinNuke (a widely available DOS tool used to remotely crash any unprotected Windows PC)
- multiple IP reassembly based attacks that include: Bonk, Boink, Netsea, Syndrop, Opentear

- application based attacks that include: Mimeflood, Octopus, Teardrop, Tentacle, and DNS spoofing

- IP reassembly

The Secure Router 2330/4134 firewall performs IP reassembly for packets to prevent IP fragment attacks.

- Stealth mode

The Secure Router 2330/4134 firewall can operate in undetectable (hidden) mode. In this case, the firewall does not send the reset packets for TCP traffic if there is no corresponding matching policy for an incoming packet.

---

## Application and URL filtering

The firewall provides the capability of filtering on certain application protocols, including:

- HTTP: allows for blocking of ActiveX, Java, jar and wild carded file extensions (such as \*.gif, \*.jpg)
- FTP: allows for blocking or allowing FTP commands like put, get, ls, mkdir, cd and pasv
- SMTP: allows for blocking or allowing SMTP operations such as hello, mail, rcpt, data, quit, send, saml, rset, vrfy, expn
- RPC: allows for denying or allowing program numbers. A remote procedure is uniquely identified by program number, version number, and procedure number. Any version and any procedure of an allowed program number is allowed.

The firewall also provides the capability to perform URL key word filtering at the global level. Web pages can be blocked based on specific key words in the URL request string.

---

## Policy-based controls

The firewall also provides the following policy-based controls:

- Policy-based connection control:

provides the capability to control the maximum number of connections that can originate from a given virtual firewall on a firewall policy basis.

- Policy-based rate control:

provides the capability to control the maximum number of packets or bytes per second on a firewall policy basis.

- Policy-based schedules:

provides the capability to control when the firewall policy is active. By associating a schedule with a firewall policy, you can determine when the firewall policy is in effect. Schedules can be configured on a recurring basis or as a one-time event. For example,

you can set a network policy of not allowing outbound FTP-PUT and mail after business hours.

- Policy-based logging:

provides logging support on a per policy basis. The logging can be on the system console, telnet console or a syslog server. This helps you to debug problems, if any.

---

## Logging and statistics

The Secure Router 2330/4134 firewall provides logging support on multiple levels. This logging can output to the system console, a telnet session, an SSH session, or a syslog server. This helps you to debug problems, if any.

Statistics are maintained on the following basis:

- Per virtual firewall basis: number of packets from and to the untrusted zone
- Per connection basis: number of bytes received and transmitted
- Global basis: number of packets received and transmitted

---

## ALG Overview

Whenever traffic is allowed to go out based on outbound policies, the firewall receives inbound traffic as a response to the outgoing traffic. In order to allow the inbound traffic to pass, the firewall creates a temporary inbound policy which expires upon the expiry of the firewall connection. This dynamic inbound policy creation requires intimate knowledge of the applications generating the traffic.

To create these policies, the stateful firewall uses Application Level Gateways (ALG). ALGs are application-aware and support dynamic port opening, providing the supported applications with the required ports to receive traffic across the firewall.

By default, all ALGs are disabled on the firewall. You can have ALG processing disabled, yet keep firewall processing. You can enable or disable one ALG, multiple ALGs, or all ALGs.

---

## Supported ALGs

The following sections describe the ALGs supported by the Secure Router 2330/4134 firewall.

## General

- FTP
- ICMP (Echo, Echo response, Destination unreachable, time exceed and source quench)
- SQLNet

## Video and streaming applications

- RTSP
- QuickTime
- RealPlayer (Real Audio and Real Video)
- H.323 (ASN1 PER encoding and decoding included)
- NetMeeting
- Intel Video Phone
- CuseeMe 5.0
- SIP

## Communication

- Internet Chat
- IRC - MIRC
- AOL Instant Messenger
- AOL enhanced chat
- ICQ2000b
- Net2Phone
- Microsoft Instant Messenger

## Security related

- PPTP
- IPsec ESP (IPsec client from internal network)
- IKE

- L2TP
- GRE

## Configurable TCP and UDP parameters for SIP ALG

Under the firewall alg tree, use the `sip-tcp` command to enable the SIP ALG on TCP port 5060 only. To enable SIP ALG on UDP port 5060 only, use the `sip` command.

With the `sip` command for UDP, you can also use the optional port parameter to specify a UDP port, other than default port 5060, to register with the SIP ALG. The default port for most SIP servers is 5060. However, some servers can listen on different ports.

For more information about configuring TCP and UDP parameters for SIP ALG, see [Configuring global ALGs](#) on page 131.

## SIP ALG interoperability with Avaya MCS clients

The Secure Router 2330/4134 firewall supports a SIP ALG that enables Avaya MCS Clients, in tandem with the MCS 5100 Server, to complete calls via application-level address translation. It also dynamically opens the necessary pinholes for media traffic to traverse the firewall.

For more details refer to [SIP ALG](#) on page 35.

---

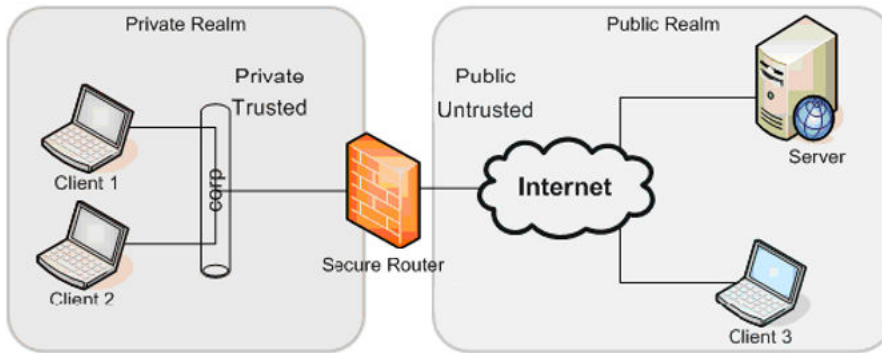
## NAT Overview

NAT gives ports on a private network access to the Internet using one or more globally unique IP addresses.

NAT contains a pool of available global addresses that are continually reused. It allows a network to use one set of network addresses internally and a different set when dealing with external networks. Internal network addresses are allocated according to internal considerations of the network. Global addresses must remain unique to distinguish between different hosts. When a packet is routed, NAT replaces the internal corporate address with a global address. As soon as the application session is over, the global address is returned to the pool and can be used by subsequent connections. NAT can also modify the source and destination port numbers.

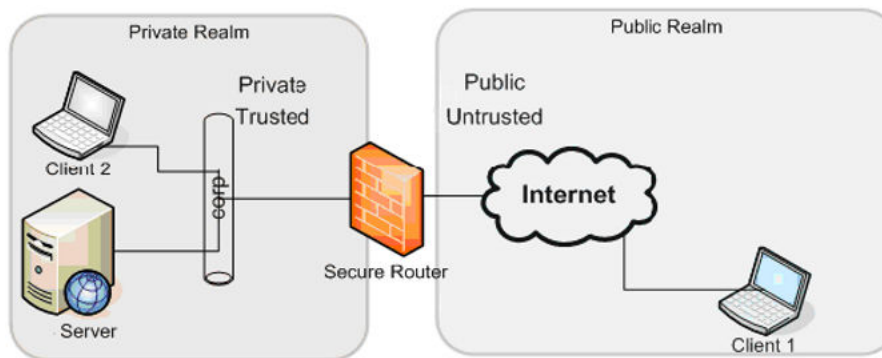
Network Address Translation (NAT) can be applied on a per policy basis. With policy-based NAT, traffic is translated only when it matches the configured firewall policy as opposed to being applied to all traffic going out an interface.

Forward NAT is when the translation happens on traffic going from the inside network (trusted) to the outside network (untrusted). With Forward NAT, NAT is applied to an outgoing firewall policy and the source IP address of the packet gets translated. The following diagram displays an example of a Forward NAT scenario.



**Figure 4: Forward NAT**

Reverse NAT is when the translation happens on traffic going from the outside network (untrusted) to the inside network (trusted). With Reverse NAT, NAT is applied to an incoming firewall policy and the destination IP address of the packet gets translated. The following diagram displays an example of a Reverse NAT scenario.



**Figure 5: Reverse NAT**

The following sections describe the various types of NAT, as well as features of NAT.

## Static NAT

Static NAT is a direct mapping of traffic from an unregistered address to a registered address on a one-to-one basis. This can be used to translate traffic going from the trusted side to the untrusted side or vice versa. It is particularly useful when a device on the inside network needs to be accessible from the outside network.

Static NAT does not change TCP or UDP port numbers.

Static NAT can change packet IP addresses.

---

## Dynamic NAT

Dynamic NAT is a dynamic many-to-many mapping of traffic from unregistered addresses to a pool of external registered IP addresses. Typically, the range of external IP addresses is less than the number of internal addresses on the trusted side. Each time a request is made from a host on the private network, the router chooses an external IP address that is currently unused, and then performs the translation. Dynamic NAT picks external IP addresses in a round robin fashion to perform the translation. Dynamic NAT can only be used for traffic initiated from an internal host.

Dynamic NAT does not change TCP or UDP port numbers.

Dynamic NAT can change packet IP addresses.

---

## PAT

Port Address Translation (PAT), also known as Network Address Port Translation (NAPT), is a form of Dynamic NAT. With PAT, multiple internal hosts can share the same public IP address. PAT maps multiple unregistered IP addresses to a single registered IP address by using different port numbers. The firewall keeps a listing of assigned port numbers to track which sessions belong to which hosts. With PAT enabled, theoretically up to 64K hosts can share a single IP address.

PAT can change the TCP/UDP port number and packet IP addresses.

## Port Restricted Cone NAT

The type of PAT supported on the Secure Router 2330/4134 is port restricted cone NAT. In this case, all requests from the same internal IP address and port are mapped to the same external IP address and port number. An external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.

---

## NAT failover for firewalls

When you specify the external address for the NAT translation, you can either specify an IP address or an interface name. If you specify an interface name for the NAT translation, packets going out through the interface are translated using the IP address of that interface. However, firewall policies do not change when an interface goes up or down. As a result, if the NAT interface goes down, NAT continues to perform the translation of internal IP addresses to the public IP address of this interface. Therefore, traffic is blackholed.



NAT failover provides a solution to this issue by allowing a primary interface (for example, T1 WAN bundle) using PAT to failover to a backup interface (for example, PPPoE or ISDN). In this case, when the primary interface is up, packets going out through the interface are translated using the IP address of the primary interface. When the primary interface goes down, the IP address of the backup interface is used for the translations, and the stale firewall connections are flushed.

To enable NAT failover, you must configure a PAT policy specifying the interface name of the primary interface. You can then specify the primary and backup interface using the **nat-failover** command.

---

## Scalability

The Secure Router 2330/4134 firewall can support the following:

- 64000 concurrent firewall connections system wide
- 25 virtual firewall zones
- 1024 policies per virtual firewall
- 30000 entries in NAT translation table

---

## Call server interoperability

This section describes Avaya Secure Router 2330/4134 interoperability with the following call servers:

- Avaya Communication Server 1000 (Avaya CS 1000)
- Avaya Multimedia Communication Server 5100 (MCS 5100)
- Live Communication Server (LCS)
- Office Communication Server (OCS)
- Sylantro and Broadsoft servers

---

## NAT traversal strategies for VoIP

There are three NAT traversal strategies available for VoIP:

- Cone NAT with Simple Traversal of UDP Networking (STUN)
- Session Initiation Protocol Application Layer Gateway (SIP ALG)
- NAT avoidance (use of VPN or other tunneling)

The following sections provide details on the first two options.

---

## NAT ACL enhancements

NAT ACL enhancements add flexibility for configuring a network Access Control List (ACL). Access Control Lists are used to filter packets going to the global NAT subsystem. A separate ACL is allowed for static translation, dynamic port translation, and dynamic address translation modules. Access Control Lists are applied to both outbound and inbound traffic for translation.

If a packet matches a permit rule, the packet enters that NAT module. If a packet matches a deny rule, it is transmitted without being modified. In the event a packet traverses all NAT ACLs without a rule match, the packet is dropped. One single NAT ACL is allowed in the Global NAT module to control access. The Global NAT ACL can be applied selectively to an interface.

---

## NAT ACL Packet Processing

The following section contains information about Packet Translation in a forwarding scenario for both incoming and outgoing packets.

### Outgoing Packet Translation

During outgoing packet translation, packets sent from a private client to a host on a public network are known as outgoing packets. NAT translation is enabled on the public interface. An ACL is applied if either the inbound interface ACL is enabled on a private interface or if the outbound interface filter is enabled on a public interface. A check is performed on the outgoing interface for NAT ability prior to the packet being sent out.

If an outgoing packet matches a static translation route, the packet is translated and sent. If ACL filters are configured for Address NAT, the following actions are taken:

- Packet is translated if it matches a permit rule
- Packet is forwarded, without being translated if it matches a deny rule

- Packet is forwarded to Address NAT module if no rule is matched.
- In the case of Dynamic Address NAT, if the module is not enabled the packet is dropped.

In the case of Dynamic Address NAT, if the module is not enabled the packet is dropped.

### Incoming Packet Translation

Packets returned to the private client from a host in a public network are known as Incoming Packets. After the packet is received, prior to route lookup, processing of address translation for the incoming packets takes place. All inbound packets are subjected to reverseACL to apply NAT translations; reverse ACL is enabled by default.

---

## Cone NAT

Most SIP applications and servers provide their own NAT Traversal strategy using STUN or a STUN-like protocol. STUN and STUN-like solutions require the NAT devices in the middle to be Cone NAT devices. The Secure Router 2330/4134 is a Cone-NAT device when configured for PAT policies. An application supported NAT Traversal strategy is preferable and more reliable to a SIP ALG.

---

## SIP ALG

The SIP ALG provides a solution for SIP-based VoIP products to operate in a firewall and NAT-enabled environment.

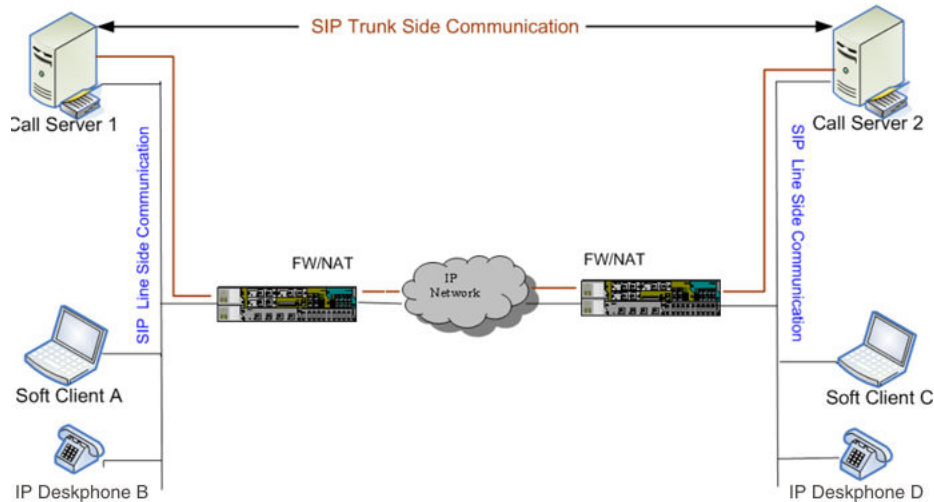
For SIP applications that do not provide an inherent NAT Traversal strategy, the SIP ALG is available for the few supported scenarios. For more information, see [Servers and scenarios supported in Secure Router 2330/4134](#) on page 40.

In a NAT environment, VoIP protocols require processing to translate IP addresses embedded in their messages. However, VoIP signalling protocols operate at Layer 5, while NAT processing typically only affects Layer 3 addressing.

The SIP Application Level Gateway provides the SIP protocol application intelligence required to complete calls using application-level address translation. The SIP ALG removes any occurrences of the internal client private IP address and port in the outgoing SIP messages and replaces these with the public IP address and port. It also monitors the message exchanges between the clients and dynamically creates and manages the pinholes required for media traffic to traverse the firewall.

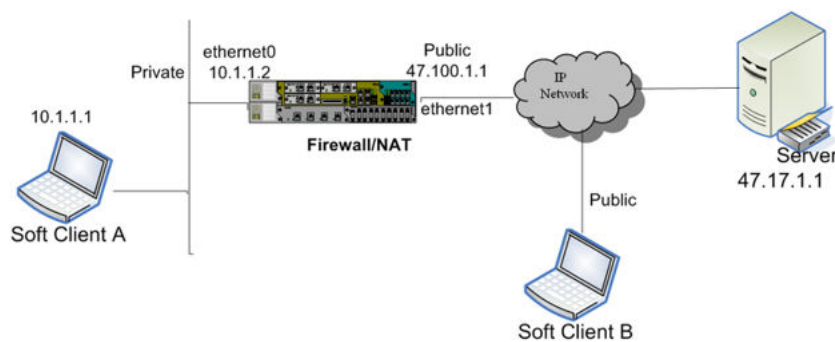
As shown in the following figure, all SIP traffic can be divided into two types depending on the origination and termination point.

- SIP Line Side: Traffic that flows between a call server and an IP set.
- SIP Trunk Side: Traffic that flows between two call servers.



**Figure 6: SIP trunk side and line side**

## Line-side SIP traffic translation



**Figure 7: PC client behind firewall/NAT**

SIP line side traffic refers to traffic that flows between a SIP call server and an IP Deskphone.

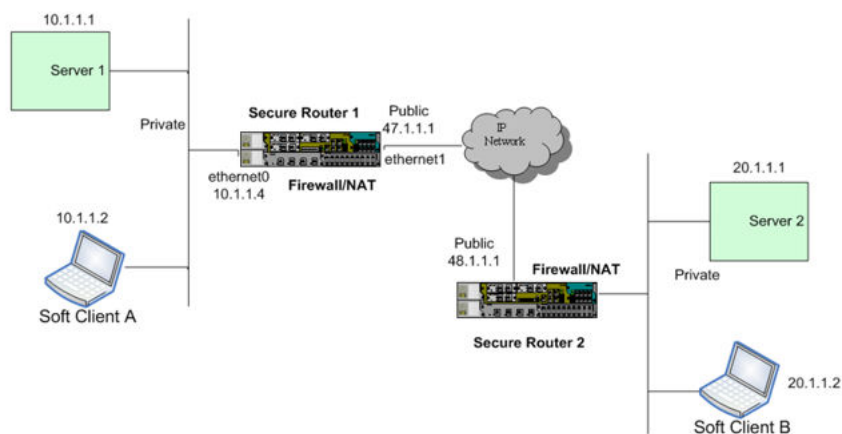
Line side traffic is triggered when the source port or destination port of the packet is the configured SIP port (default is 5060). The SIP ALG translates the private IP address and port of the internal client in the outgoing SIP message body (which defines connection and media parameters). For SIP messages received, the SIP ALG restores the private IP address of the internal client. The SIP ALG also dynamically opens the necessary pinholes for media traffic to traverse the firewall.

## Line side SIP traffic configuration tips

The following tips apply if the client is in the private network.

- If only Firewall is enabled, the default firewall policies are sufficient.
- If both Firewall and NAT are enabled, configure an outgoing policy in the 'corp' map to NAT the outbound traffic to the public nat-ip address

## Trunk-side SIP traffic translation



**Figure 8: SIP trunk side configuration**

Trunk side traffic refers to traffic that flows between two call servers.

When a phone attempts to make a call to a phone that is not registered on the same call server, the local call server must relay that request to the next-hop call server. The forwarded request from the local call server to the next-hop call server is trunk traffic.

SIP trunking can introduce further complications with firewalling and NAT. Specifically, the SIP signaling information is exchanged between the SIP servers, while the media session traffic flows between the two phones. As a result, the SIP ALG must translate multiple private addresses within one packet. For example, when the SIP ALG encounters the INVITE message, the SIP header contains the IP address of the call server and the SDP session description contains the IP address of the phone. However, NAT rules generally control only one-to-one address and port translations

To workaroud these issues, the SIP ALG can implement a form of Proxy NAT.

With Proxy NAT, the SIP ALG performs multiple translations within a single packet. It performs a Static NAT translation for the SIP header, and a NAPT translation for the SIP message body (SDP). This results in a single firewall connection between the two call servers on port 5060, for all SIP signaling, and multiple RTP connections for media traffic between the phones.

In this scenario, there can be only one server behind the Firewall/NAT, supported by a single proxy NAT command.

## Trunk side SIP traffic configuration tips

The following tips apply if the server is in a private network.

- If only Firewall is enabled, configure an inbound policy in 'corp' map to allow packets on port 5060
- If both Firewall and NAT are enabled, configure an inbound policy in 'corp' map to allow packets on port 5060 and reverse NAT to the MCS server private address.

---

## NAT Hairpinning

Hairpinning allows two endpoints on the internal side of the NAT to communicate even if they only use external IP addresses and ports.

If two hosts are behind a NAT and exchanging traffic, the NAT device may allocate a public address and port for each of them to use. If the two hosts communicate with each other via their public NAT addresses, hairpinning allows the NAT device to receive and return the packets on the same interface while translating the address and port mappings.

To implement hairpinning on the Secure Router 2330/4134, you must have one of the following NAT traversal strategies in place:

- STUN
- SIP-ALG

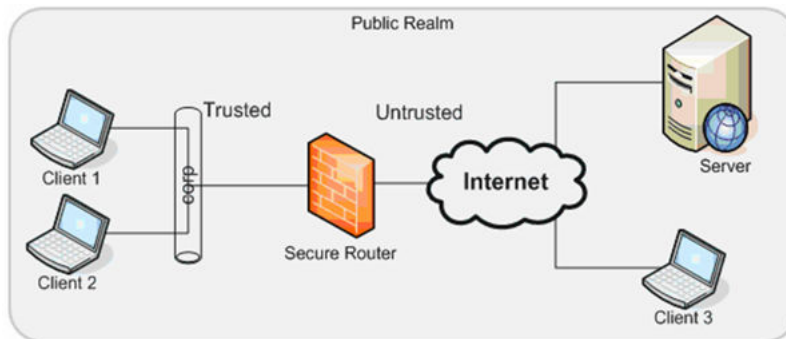
A limitation with Secure Router 2330/4134 is that hairpinning and self policies are mutually exclusive on the router. You must make a choice between either allowing hairpinning or allowing inbound connections from the Internet to a self IP.

---

## Scenarios

The following figures show scenarios describing the use of firewall and NAT with SIP server topologies.

## Firewall only

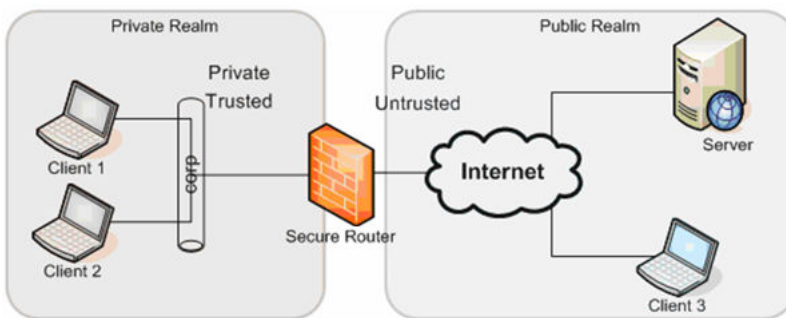


**Figure 9: Firewall only**

In this scenario:

- All IP Addresses are public
- Client1, Client2 and Client3 register with the server
- Line Side Configuration

## Forward NAT



**Figure 10: Forward NAT**

In this scenario:

- The server is in the public realm
- Private and public realm clients register with the server
- Line Side Configuration with NAT

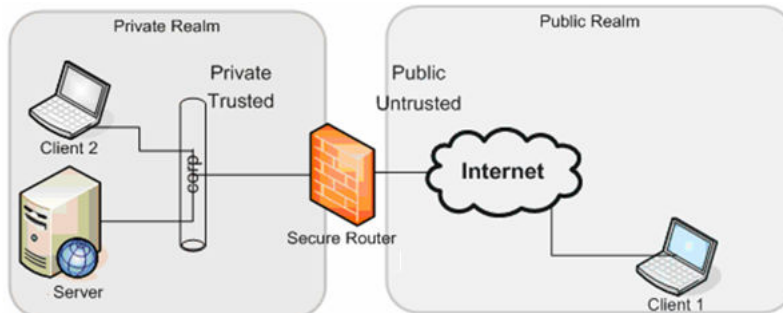
---

## Forward NAT with Hairpinning

Forward NAT with Hairpinning is similar to the preceding Forward NAT scenario, providing feature support between clients in the private realm.

---

## Reverse NAT



In this scenario:

- The server is in the private realm
- Configure an inbound firewall policy in the trusted zone to allow the clients in the public realm to register and communicate with clients in the private realm
- Enable Proxy-NAT command with Server-Ip Address in Private Realm

---

## Servers and scenarios supported in Secure Router 2330/4134

The following sections describe the SIP servers and scenarios that are supported with the Secure Router.

---

### Multimedia Communication Server (MCS)

- SIP-Line Side: Firewall Only, Forward NAT, Forward NAT with Hairpinning, and Reverse NAT
- SIP-Trunk Side: Firewall Only, Forward/Reverse NAT with Private Servers
- SIP ALG supports audio/video call, instant messaging, whiteboard sharing, file sharing, call transfer, call redirect (not auto-redirect) with MCS



---

## Live Communication Server (LCS)

- SIP-Line Side: Firewall Only, Forward NAT, and Forward NAT with Hairpinning
- SIP ALG supports voice calls, video conversation, IM, and application sharing with LCS

---

## Office Communication Server (OCS)

- OCS client uses STUN for NAT traversal.
- Disable SIP ALG for all OCS scenarios.
- SIP-Line Side: Firewall Only, Forward NAT, and Forward NAT with Hairpinning

---

## Sylantro and Broadsoft servers

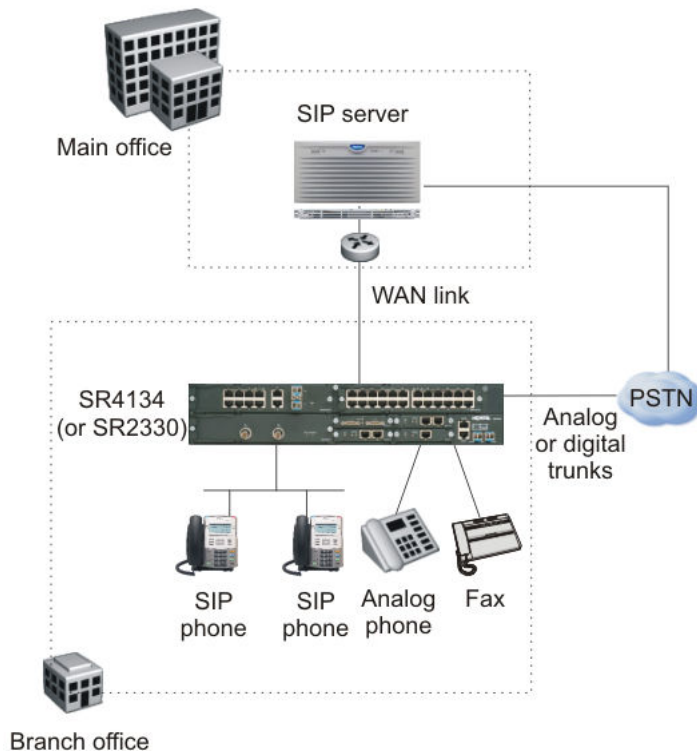
- SIP-Line Side - Firewall Only, Forward NAT, and Forward NAT with Hairpinning
- SIP ALG supports voice calls

---

## Avaya Communication Server 1000

The Avaya CS 1000 series call servers and Avaya IP Deskphones implement a STUN-aware protocol for UNISTim, and can therefore support NAT Traversal.

## SIP ALG support for Media Gateway and SSM in Secure Router



**Figure 11: Secure Router topology with Media Gateway**

- Support of public server only
- Media Gateway can have a public IP address in the Firewall only scenario
- Media Gateway and SSM can have private IP addresses in Forward NAT scenario

The SIP Server at the main office can be one of the following:

1. Avaya CS 2100
2. CS2000
3. Sylantro
4. BroadSoft
5. A2E

Note that servers having Back-Back User Agent capability can interoperate with the Secure Router.

---

## Firewall only with Media Gateway

- Configure firewall policy to allow sip-udp traffic
- Bind Media Gateway to public IP

---

## Forward NAT With Media Gateway

- Configure firewall policy with NAT-IP as untrusted interface in Internet Zone
- Bind Media Gateway to private IP
- Provide support of Peer-Peer RTP media between clients in private realm by enabling `sip-p2p-media` under Global ALGs
- Support of SIP Over UDP as transport

---

## Forward NAT with SSM

- Configure firewall policy with NAT-IP as untrusted interface in Internet Zone.
- Bind SSM to Private IP
- Enable Proxy-NAT command under global ALGs with IP address as the SSM Bind Address
- Support of SIP Over UDP as transport.

---

## Standards compliance

The Secure Router 2330/4134 firewall implementation complies with the following RFCs:

- RFC 1579, Firewall-Friendly FTP
- RFC 1858, Security Considerations for IP Fragment Filtering
- RFC 2979, Behavior of and Requirements for Internet Firewalls
- RFC 3093, Firewall Enhancement Protocol (FEP) – Supports HTTP
- RFC 1948, Defending Against Sequence Number Attacks
- RFC 2827, Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing

- RFC 1631, The IP Network Address Translator (NAT)
- RFC 2663, IP Network Address Translator (NAT) Terminology and Considerations – Except support for twice NAT and RSIP
- RFC 2694, DNS Extensions to Network Address Translators (DNS\_ALG)

# Chapter 4: Packet filter fundamentals

The central idea of a packet filter is that a set of rules are defined to monitor inbound connections at the network layer level. As long as the packet conforms to the defined rules, the packet is allowed to pass. If the packet does not conform to the defined rules, it is dropped.

With the Avaya Secure Router 2330/4134, the packet filter feature provides stateless, interface-based packet filtering as an alternative to the stateful firewall. It also provides IPv6 and MAC packet filter functionality to complement the IPv4-only stateful firewall.

The Secure Router 2330/4134 packet filter examines each packet on the interface to determine whether to permit or drop the packet, based on the criteria specified within user-configured access lists. This control can restrict network traffic and restrict network use for certain users or devices.

The Secure Router 2330/4134 supports three packet filter types; IPv4, IPv6, and MAC.

Some Secure Router 2330/4134 interfaces are equipped with hardware accelerated switching components that enforce packet-filters. For interfaces that are not equipped with these components, packet filters are software-enforced. Hardware enforced packet filters have slightly different behaviors than software enforced packet filters.

This chapter describes the packet filter capabilities on the Secure Router 2330/4134. It also describes the differences between hardware and software enforced packet filters, as well as the implementation differences between the Secure Router 4134 and Secure Router 2330 chassis.

---

## Packet filters on specific Secure Router chassis

The Secure Router 2330 and Secure Router 4134 both support software enforced packet filters and hardware enforced packet filters. Hardware enforced packet filters help reduce CPU workload and improve the processing speed for Ethernet-based packets. However, hardware enforced packet filters are only supported on specific interfaces.

The following sections describe packet filter enforcement characteristics on the Secure Router 4134 and the Secure Router 2330.

---

### Packet filters on the Secure Router 4134

On the Secure Router 4134, hardware-enforced packet filters are supported on module Ethernet ports only. On chassis Ethernet ports (ports 0/0 to 0/4), WAN interfaces and VLANs, the packet filters are software enforced.

With VLAN packet filters, the port members must be chassis Ethernet ports. If the VLAN has any module Ethernet port members, packet filtering is not supported.

Module Ethernet ports operate at a higher throughput than the CPU controlled chassis Ethernet ports and WAN interfaces. With the higher throughput on module Ethernet ports, hardware packet filter enforcement preserves switching operation speed on the Secure Router 4134.

The following table shows the Secure Router 4134 packet filter enforcement type to port type relationship.

**Table 1: Secure Router 4134 packet filter enforcement type to port type relationship**

Port type	Packet filter enforcement type	Supported packet filters	Direction
WAN interfaces	Software-enforced	MAC, IPv4, and IPv6	egress and ingress
Chassis Ethernet ports (0/x)	Software-enforced	MAC, IPv4, and IPv6	egress and ingress
IP-VLAN (with all members in chassis Ethernet ports)	Software-enforced	MAC, IPv4, and IPv6	egress and ingress
IP-VLAN (with any member in module Ethernet ports)	Not supported	N/A	N/A
Module Ethernet ports	Hardware-enforced	MAC, IPv4, and IPv6	ingress only

## Secure Router 4134 packet filter feature limitations

The following packet filter limitations exist with the Secure Router 4134:

- Hardware-enforced packet filters support IPv4, IPv6, and MAC packet filters on ingress traffic only. You can apply one packet filter of each type on each interface.
- Software-enforced packet filters support one IPv4 and one IPv6 packet filter for either ingress or egress traffic. You can apply one packet filter of each type on each interface.

## Secure Router 4134 hardware-enforced packet filter rule limitations

With hardware packet filter enforcement, the following packet filter rule limitations exist on module Ethernet ports:

- Packet filter rules cannot be applied as outbound rule lists.
- Packet filter rules cannot contain TCP or UDP port ranges and the following operators are not supported:
  - less than (<)
  - greater than (>)

- less than or equal to ( $\leq$ )
- greater than or equal to ( $\geq$ )
- Packet filter rules cannot include the *log on* parameter.
- Packet filter rules do not log information or display statistics when using the **show <proto> packet-filter-stats** command.

Software-enforced packet filters do not have these limitations.

---

## Packet filters on the Secure Router 2330

On the Secure Router 2330, hardware enforced packet filters are supported on chassis Ethernet ports only. On WAN and VLAN interfaces, the packet filters are software enforced.

Hardware packet filter enforcement helps reduce CPU workload and improves Secure Router 2330 speed for processing Ethernet-based packets.

The following table shows the Secure Router 2330 packet filter enforcement type to port type relationship.

**Table 2: Secure Router 2330 packet filter enforcement type to port type relationship**

Port type	Packet filter enforcement type	Supported packet filters	Direction
WAN interfaces	Software-enforced	IPv4 and IPV6 only	egress and ingress
IP-VLAN interfaces	Software-enforced	IPv4 and IPV6 only	egress and ingress
Chassis Ethernet ports	Hardware-enforced	MAC, IPv4, and IPV6	ingress

## Secure Router 2330 packet filter feature limitations

The following packet filter limitations exist with the Secure Router 2330:

- Hardware-enforced packet filters support MAC, IPv4 and IPv6 packet filters on ingress traffic only. On each interface, you can apply one packet filter of each type.
- Software-enforced packet filters support IPv4 and IPv6 packet filters on egress traffic only. On each interface, you can apply one packet filter of each type. The Secure Router 2330 does not support software-enforced MAC packet filters.

## Secure Router 2330 hardware-enforced packet filter rule limitations

With hardware packet filter enforcement, the following packet filter rule limitations exist on the Secure Router 2330 Ethernet ports:

- Packet filter rules cannot be applied as outbound rule lists.
- Packet filter rules cannot contain TCP or UDP port ranges and the following operators are not supported:
  - less than (<)
  - greater than (>)
  - less than or equal to (<=)
  - greater than or equal to (>=)
- Packet filter rules cannot contain the *log on* parameter.
- Packet filter rules do not log information or display statistics when using the **show <proto> packet-filter-stats** command.

Software-enforced packet filters do not have these limitations.

---

## Packet filter relationship to other Secure Router 2330/4134 features

The Secure Router 2330/4134 hardware and software enforced packet filter processes have interactions and exclusions with the following features:

- firewall
- virtual private network (VPN)
- packet capture (PCAP)
- quality of service (QoS)

---

## Packet filter relationship to firewall and VPN

You cannot configure software enforced packet filters and firewall or VPN on the same port. You can configure hardware enforced packet filters and firewall or VPN on the same port, but there are limitations depending on the Secure Router chassis you are using.

The following table shows the firewall and VPN relationship to packet filter enforcement types on Secure Router 4134 ports.



**Table 3: Secure Router 4134 firewall and VPN to packet filter enforcement type relationship**

Port type	Packet filter enforcement type	IPv4 Packet filter + firewall or VPN	IPv6 Packet filter + firewall or VPN	MAC Packet filter + firewall or VPN
WAN interfaces	Software-enforced	No	Yes (can be listed in trusted or untrusted zones)	Yes (can be listed in trusted or untrusted zones)
Chassis Ethernet ports (0/x)	Software-enforced	No	Yes (can be listed in trusted or untrusted zones)	Yes (can be listed in trusted or untrusted zones)
IP-VLAN (with all members in chassis Ethernet ports)	Software-enforced	No	Yes (can be listed in trusted or untrusted zones)	Yes (can be listed in trusted or untrusted zones)
IP-VLAN (with any member in module Ethernet ports)	N/A	N/A	N/A	N/A
Module Ethernet ports	Hardware-enforced	Yes (can be listed in trusted zones only)	Yes (can be listed in trusted zones only)	Yes (can be listed in trusted zones only)

The following table shows the firewall and VPN relationship to packet filter enforcement types on Secure Router 2330 ports.

**Table 4: Secure Router 2330 packet filter enforcement type to port type relationship**

Port type	Packet filter enforcement type	IPv4 Packet filter + firewall or VPN	IPv6 Packet filter + firewall or VPN	MAC Packet filter + firewall or VPN
WAN interfaces	Software-enforced	No	Yes (can be listed in trusted or untrusted zones)	N/A (software-enforced MAC packet filter not supported on SR2330)
IP-VLAN interfaces	Software-enforced	No	Yes (can be listed in trusted or untrusted zones)	N/A (software-enforced MAC packet filter not supported on SR2330)
Chassis Ethernet ports	Hardware-enforced	Yes (can be listed in trusted	Yes (can be listed in trusted	Yes (can be listed in trusted

Port type	Packet filter enforcement type	IPv4 Packet filter + firewall or VPN	IPv6 Packet filter + firewall or VPN	MAC Packet filter + firewall or VPN
		or untrusted zones)	or untrusted zones)	or untrusted zones)

---

## Packet filter relationship to PCAP

You can optionally configure PCAP with rules similar to packet filter rules to limit the traffic that an interface captures. When you enable hardware enforced packet filters and PCAP on an interface, both rule sets are combined into a single list in the secure router hardware. The hardware uses parallel lookups to match packet filter and PCAP rules, with certain actions maintaining a higher precedence. For example, a drop action has the highest precedence, so if a packet filter resulting action is deny, PCAP actions are ignored. The system will perform PCAP only if the matching packet filter is a permit rule.

---

## Packet filter relationship to QoS

QoS uses rules similar to packet filter rules to classify ingress traffic on an interface. When you enable hardware enforced packet filters and QoS, both rule sets are combined into a single list in the Secure Router hardware. The hardware uses parallel lookups to match packet filter and QoS rules, with certain actions maintaining a higher precedence. For example, a drop action has the highest precedence, so if a packet filter resulting action is deny, QoS actions are ignored. The system will perform QoS only if the matching packet filter is a permit rule.

## Maximum allowable filter rules on Ethernet modules

On each Ethernet module, the TCAM limits the number of allowable packet filter rules. The TCAM provides 896 entries that are shared between Ethernet module QoS and the packet filters (regardless of the number of ports on the module). This limits the number of filter rules that can be applied on the Ethernet modules.

In addition, a packet filter rule can consume more than one entry in the TCAM depending on the packet match qualifiers.

---

## Available packet filters

The following sections provide more detail on packet filters available with the Secure Router 2330/4134.

---

## IPv4 packet filters

The IPv4 packet filter performs filtering of IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP). The available IPv4 filtering parameters are as follows:

- Source and destination IP address
- Protocol, including TCP, UDP, ICMP, or IGMP.
- Source and destination port numbers: filters can use port numbers to restrict the network traffic to a limited set of services, each of which is associated with a well-known port.
- TCP flags: filters can check the TCP ACK and SYN bits, indicating whether a new connection is being set up or not. This is particularly useful to allow TCP sessions being initiated only from one side.
- ICMP message types: filters can restrict ICMP traffic to a limited set of message types. For example, the Echo Request packets can be refused
- IGMP message types: filters can restrict IGMP traffic to a limited set of message types. For example, the Group Membership Query packets can be refused.
- User data: filters can restrict traffic based on user data found in the (protocol-specific) data part of the IP packet.
- Network Devices: filters can act differently for different network devices through which a packet is received or is going to be sent, such as external or internal interfaces.
- Date and time: filters can limit some types of network traffic to office hours, for example.
- Fragments - filters can restrict non-initial fragmented packets where the fragment extension header contains a non-zero fragment offset. The fragments keyword is an option only if the operator [port-number] arguments are not specified.

---

## IPv6 packet filters

The IPv6 packet filter performs filtering of IPv6 traffic, based on the rules configured. The available IPv6 packet filtering parameters are as follows:

- Protocol: filters can be configured based on one of the keywords tcp, udp, icmp, ipv6, or on an integer in the range from 0 to 255 representing an IPv6 protocol number.
- Source IPv6 prefix/prefix-length: filters can restrict traffic from a source IPv6 network or class of networks. The IPv6 address must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.
- Destination IPv6 prefix/prefix-length: filters can restrict traffic with a specified destination IPv6 network or class of networks. The IPv6 address must be in the form documented in

RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.

- Source port or destination port: filters can use source or destination port numbers to restrict traffic. The port number value must be a decimal value between 0 and 65535.
- ICMP type: filters can use the ICMP message type to filter the ICMP packets. The type is a number from 0 and 255.
- ICMP code: filters can use the ICMP message code to filter the ICMP packets if specified along with ICMP message type. The code is a number from 0 and 255.
- TCP flags: filters can use the TCP flags for packet filtering. Available options are:
  - Keyword established can be used to match already established connections. The non-matching case is that of the initial TCP datagram to form a connection.
  - Keywords fin, syn, ack, psh, rst and urg can be used to match the corresponding TCP header flags. You can specify multiple TCP flag keywords but they should be specified with a comma (,) in between and with no space. For example, if “flags syn” is specified, only TCP packets with SYN flag alone are matched; a TCP packet with both SYN and ACK flag set would not be matched.
- DSCP value: Matches a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63.
- Flow-label: filters can restrict traffic that matches a flow label value in the Flow Label field of each IPv6 packet header. The acceptable range is from 0 to 1048575.
- Routing - filters can restrict source-routed packets that match the routing extension header within each IPv6 packet header.

---

## MAC packet filters

The MAC packet filter performs filtering based on MAC header information. The basic MAC filtering function is to filter based on source MAC address and destination MAC address with mask. In addition to MAC address information, MAC packet filter can filter based on Ethernet type, VLAN ID, and COS. Typically, the main object of MAC packet filters is to filter non-IP packets within a LAN area. However, MAC packet filters can also filter MAC information of IP packets to support VLAN.

The MAC packet filter can only be applied to module Ethernet interfaces in the ingress direction.

---

## Packet filters for management services

Interface packet filters can only be applied and enforced on individual Secure Router interfaces, and are therefore inefficient for some applications. For example, if a network

administrator intends to use an interface packet filter to block a particular management service, such as a Telnet server, it is necessary to create and apply a specific packet filter rule list on each active Secure Router interface separately. This process can be time consuming and subject to errors.

To simplify the blocking of management services, the Secure Router supports packet filters for management services, which allow you to create and apply a packet-filter rule list globally on the router, independent of the router interfaces. Because the intent of the feature is to provide control of management services only, this filter is applicable for local traffic only, be it inbound, outbound, or both, depending on the configuration.

### **Configuration considerations and limitations**

You must be aware of the following considerations and restrictions before you create and apply management services packet filters on the Secure Router:

- Management services packet filters are supported only with IPv4 and IPv6 filtering. You cannot apply a MAC packet filter rule list as a management services packet filter.
- You can apply both IPv4 and IPv6 management services packet filters simultaneously on a Secure Router.
- If you apply both an interface packet filter and a management services packet filter for inbound traffic processing, the interface packet filter takes precedence over the management services packet filter.
- If you apply both an interface packet filter and a management services packet filter for outbound traffic processing, the management services packet filter takes precedence over the interface packet filter.
- Unlike interface packet filters, management services packet filters can function on interfaces configured with the firewall.
- By default, an interface packet filter rule list includes an implicit deny-all rule and traffic is allowed through the filter only if you specify a rule that permits that particular traffic. Because management services packet filters are applied globally, this behavior could lock all users out of router access. Therefore, management services packet filters include an implicit permit-all rule by default. In this case, traffic is denied only if you specify a rule that denies that particular traffic.
- The logging of filter hits or access list violations with management services packet filters operates the same as for interface packet filters. The Secure Router does not support IPv6 logging.

---

## **Packet filter logging**

Logging is available for the software enforced packet filters. However, logging is not supported for hardware-enforced packet filters.

---

## Packet filter scalability and capacities

This section compares scalability and capacity considerations for the Secure Router 4134 and the Secure Router 2330.

---

### Secure Router 4134

The following scalability and capacity considerations apply to the Secure Router 4134:

- Each Secure Router 4134 supports a maximum total of 50 rule sets.
- The Secure Router 4134 supports a maximum of 896 packet filter rules for each rule set. (Including MAC, IPv4, and IPv6 rules.)

With hardware-enforced packet filters, the TCAM limits the number of allowable packet filter rules. The TCAM provides 896 entries that are shared between QoS and the packet filters (regardless of the number of ports on the module). This limits the number of filter rules that can be applied on the hardware-enforced packet filters.

---

### Secure Router 2330

The following scalability and capacity considerations apply to the Secure Router 2330:

- Each Secure Router 2330 supports a maximum total of 50 rule sets.
- The Secure Router 2330 supports a maximum of 128 packet filter rules for each rule set.
- Each Secure Router 2330 supports a maximum total of 512 packet filter rules. (Including MAC, IPv4, and IPv6 rules.)

The 512 packet filter rule maximum includes the following:

- a maximum of 256 MAC packet filter rules
- a maximum of 128 IPv4 packet filter rules
- a maximum of 128 IPv6 packet filter rules

With hardware-enforced packet filters, the TCAM limits the number of allowable packet filter rules. The TCAM provides 512 entries that are shared between QoS and the packet filters (regardless of the number of ports on the module). This limits the number of filter rules that can be applied on the hardware-enforced packet filters.

---

## Packet filter configuration considerations

When configuring packet filters, be aware of the following configuration considerations:

- At the end of every rule list is an implied "deny all traffic" statement. Therefore, all packets not explicitly permitted by filtering rules are denied. An empty packet filter list drops all packets. A packet that does not match any rule in a packet filter list is dropped. Therefore, it is important that each filter list contain at least one "permit" statement.
- The order in which you enter the filtering rules is important. As the Secure Router 2330/4134 is evaluating each packet, the OS tests the packet against each rule statement sequentially. After a match is found, no more rule statements are checked. For example, if the first rule you create is a statement that explicitly permits all traffic, all traffic is passed since no further rules are checked.
- The Secure Router 2330/4134 supports re-ordering of filter commands through insert and delete commands.
- You can configure rules with expire options. In this case, the rule expires after the specified number of seconds. The timer starts when the rule is attached to an interface.
- Packet Filter lists can be applied to more than one interface. Start with a separate packet filter list per interface, until the list is proven.
- Start with a minimum rule list and add to it as necessary.

---

## Packet filter limitations and restrictions

In addition to any previously mentioned packet filter limitations, the following feature limitations and restrictions exist with the Secure Router 2330/4134:

- The Secure Router 2330/4134 packet filter system does not support a redirect action to force traffic forward to a named next-hop.
- The Secure Router 2330/4134 packet filter system does not support a Network Address Translator (NAT) action.
- Logging is supported only on software enforced packet filter sets.
- Statistics are supported only on software enforced packet filter sets.
- The Secure Router 2330/4134 supports a limited number of filter rules.
- Throughput performance speed is impacted on ports with packet filters enabled.
- A busy Secure Router 2330/4134 drops packet filter logging events before sacrificing traffic forwarding. Some traffic can get through a busy system without being logged.

- You can apply MAC packet filters only as ingress filters.
- When you apply packet filters to VLANs in a Secure Router 4134, all ports in the VLAN must be chassis Ethernet ports.

---

## Packet filter troubleshooting

- Error “Failed to register packet filter module” usually occurs when an IP/IPv6 address has not been configured on the interface.
- ‘Expire’ rules start to expire as soon as the Packet Filter is associated with an interface. The expiring rule is deleted from the packet filter list when time runs out.



# Chapter 5: IPsec VPN fundamentals

Internet Protocol (IP) packets have no inherent security. It is relatively easy to forge the addresses of IP packets, modify the contents of IP packets, replay old packets, or inspect the contents of IP packets in transit.

IP Security (IPsec) is a protocol suite designed to provide protection for IP packets. IPsec offers the following protective services for IP packets:

- Authentication (of data origin)
- Data Integrity
- Confidentiality (of data content)
- Access control
- Replay protection

IPsec can protect packets between hosts, between security gateways (for example, routers or firewalls), or between hosts and security gateways.

IPsec uses symmetric ciphers (encryption algorithms such as DES, 3DES, and AES ) to provide confidentiality services and keyed MAC (hash algorithms such as MD5, and SHA1) to provide data integrity services. Both the encryption and hash algorithms require shared keys between the end points of the secure communication.

The shared keys for the symmetric cryptographic algorithms used by IPsec can be manually configured, but this is not easily managed for multiple IPsec connections. To provide a scalable solution, a standard (key management) method has been defined to dynamically authenticate peers, negotiate security services, and generate shared keys. This protocol is called Internet Key Exchange (IKE).

IPsec based virtual private network (VPN) operates in the network layer. Based on the policy defined, it secures individual IP packet. So, it is transparent to the higher layer applications.

There are two basic types of VPN, each with an associated set of business requirements:

- Site-to-Site VPN
- Remote access VPN

---

## Site-to-Site VPN

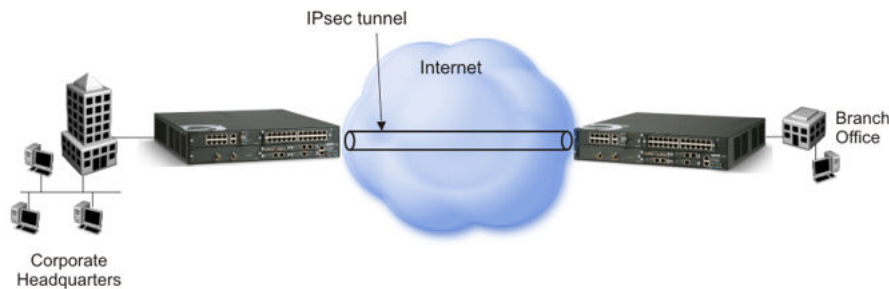
In its basic configuration, site-to-site VPN connects two remote offices or a branch office to headquarters. In this case, each site is connected to the Internet through a secure router. The objective of the site-to-site VPN is to create a secure tunnel between the two secure routers through the Internet. All the network traffic from each site to the Internet traverses the local

secure router. Each secure router monitors the network traffic and, based on the defined policy, secures the packets destined to the other secure site.

Traffic headed to any other locations on the Internet is left unprotected. In this way, the VPN transparently secures the traffic generated by the networks behind it.

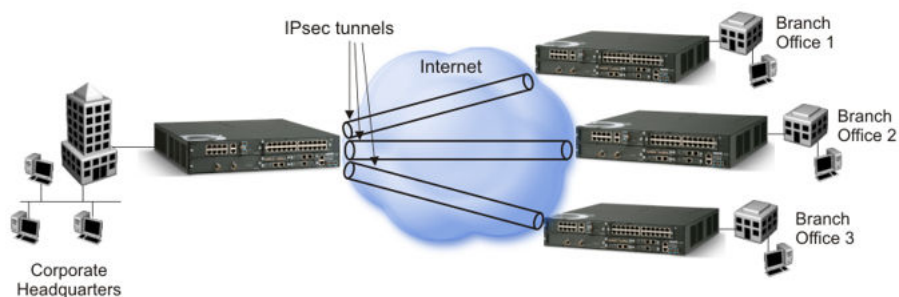
Site to site VPNs on the Avaya Secure Router 2330/4134 support any of the following topologies

- One to One: provides one-to-one connectivity between two different sites.



**Figure 12: One to one**

- One to Many: connects one site to several other sites. A typical application is to connect a headquarters with several branch offices. The headquarters acts as hub and the connections from headquarters to branch offices act as the spokes. In this case, communication between two spokes is carried through the hub.



**Figure 13: One to Many Hub and spoke VPN**

- Many to Many: connects many sites to many other sites in a mesh topology. In this case, communication is carried directly from one site to the next.

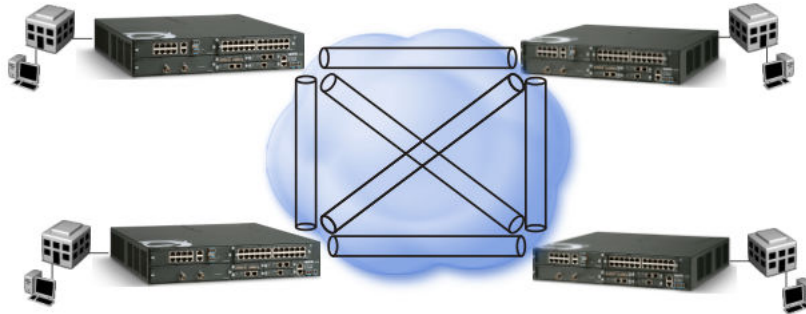


Figure 14: Mesh VPN

## Tunnel failover using static weighted tunnels

The static weighted tunnels feature provides IPsec VPN tunnel failover capability for a branch office. This feature allows an alternate tunnel to be available if the primary tunnel fails.

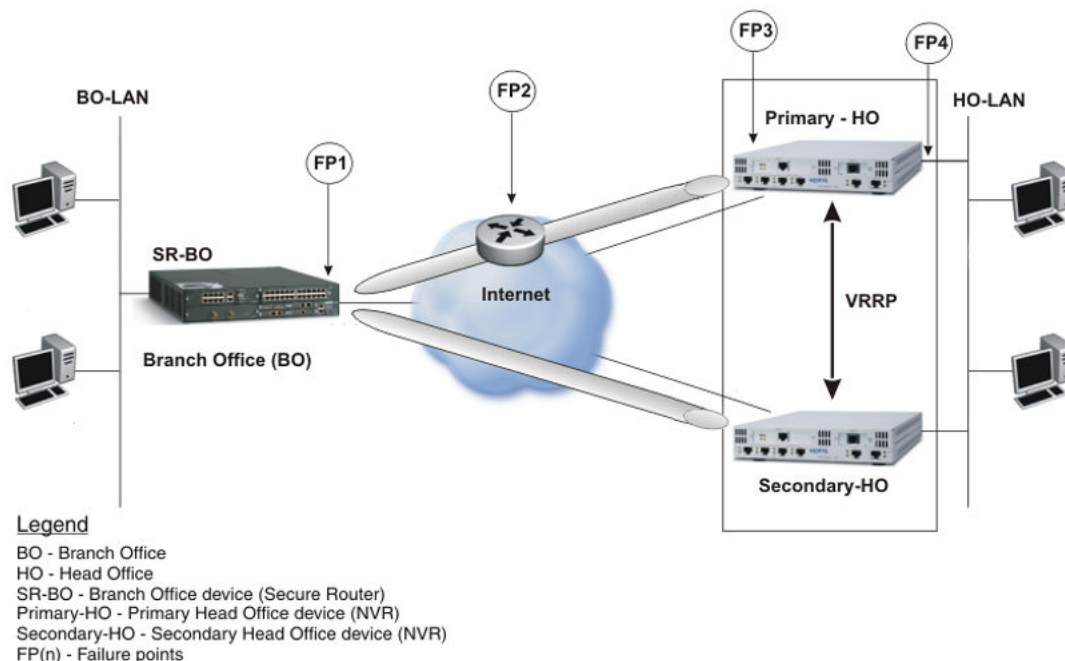
Static failover requires two VPN tunnels to be created between branch office and head office. One tunnel is labelled as **primary** and the other as **backup**. Each of these tunnels connects to an unique peer (gateway IP address) in the head office.

If the primary branch office tunnel fails, traffic is directed to the secondary tunnel. When the primary tunnel is restored, traffic will resume in the primary branch office tunnel.

The primary and backup tunnels in the branch office have one to one relationship, where one backup tunnel is configured for each primary tunnel. It is mandatory to configure the primary tunnel as **nailed-up**. The nailed-up configuration attempts to keep the tunnel always-on, which is the only way traffic can be restored back to the primary tunnel after failover.

The Secure Router uses Dead Peer Detection (DPD) to detect failure of the primary tunnel. If the the primary tunnel fails, traffic from the branch office automatically flows into the backup tunnel. Because the primary tunnel is configured to be **nailed-up**, it persistently attempts to build a tunnel with the head office. When the primary comes up, the traffic is moved back to the primary tunnel.

The following diagram shows a head office connected to a branch office using primary and secondary IPsec tunnels.



**Figure 15: Tunnel failover - static weighted tunnels**

The following table lists the supported combinations available with the Secure Router for tunnel failover on primary and secondary branch office tunnels:

**Table 5: Tunnel failover — supported combinations**

Primary	Secondary
Static Weighted – Nailed up	Static weighted – On demand
Static Weighted – Nailed up	Static weighted – Nailed up
Dynamically routed	Dynamically routed

## Tunnel failover using round robin DNS

When you configure IKE and IPsec site-to-site policies, you must identify the peer with which to negotiate. Current standards for implementing IKE and IPSec site-to-site policies require support for naming a network peer by a specific IP address only. The Secure Router 2330/4134 supports a standard exceeding feature that allows you to name a peer by a Domain Name Server (DNS) name. With this capability, DNS round robin replies can provide a form of failover and load balancing.

### Failover using round robin DNS

Round robin DNS can provide IPsec VPN tunnel failover for branch office peers. This functionality is dependant on the availability of multiple head office peers, and a DNS server running a round robin algorithm for name resolution among the head office peers.

When the initiator at the branch office performs a DNS query, the DNS server returns a list of IP addresses mapped to the same DNS name. The Secure Router always selects the first IP address in the list and establishes a tunnel. If the tunnel fails, the initiator must reestablish the tunnel and send a new DNS query. In this case, the DNS server again returns a full IP address list, but in a different order. The Secure Router then selects the first address in the new list and establishes a tunnel with the backup router, achieving failover.

If the first selected peer IP address is unreachable, the Secure Router continues to attempt establishing a tunnel until the negotiation succeeds, and performs DNS lookups until a working peer IP address is returned.

### **Load balancing using round robin DNS**

In addition to the failover capability, you can use round robin DNS to achieve simple load balancing between two head office routers by mapping both head office router IP addresses to the same DNS name.

### **Tunnel failover using round robin DNS feature considerations and limitations**

The following considerations and limitations apply when you use a DNS name to identify specific peers in IKE and IPSec policies:

- Tunnel failover using round robin DNS requires multiple tunnel endpoints at the head office.
- For gateway address resolution, this failover assumes the DNS server can provide a unique tunnel endpoint for every DNS query from the branch-office.
- The branch office Secure Router seeks to connect to a head office endpoint serially. This increases down time more than using preset backup tunnels.
- A branch office tunnel that uses a domain name to identify a peer can only operate in initiator-only mode.

The following diagram shows a site-to-site network using round robin DNS to provide tunnel failover. The branch office Secure Router (SR-BO) is configured with a security policy database (SPD) policy for the head office (HO) and an associated IKE policy. SR-BO is configured with the DNS name of the HO security gateway. HO has two tunnel endpoints to terminate the BO connections and the IP addresses of these endpoints are associated with the DNS name used by the BO. The DNS server, which is external to SR-BO, is configured to service name resolution queries using a round robin algorithm, to ensure that each query for a given DNS name is serviced with a different IP address than the previous query.

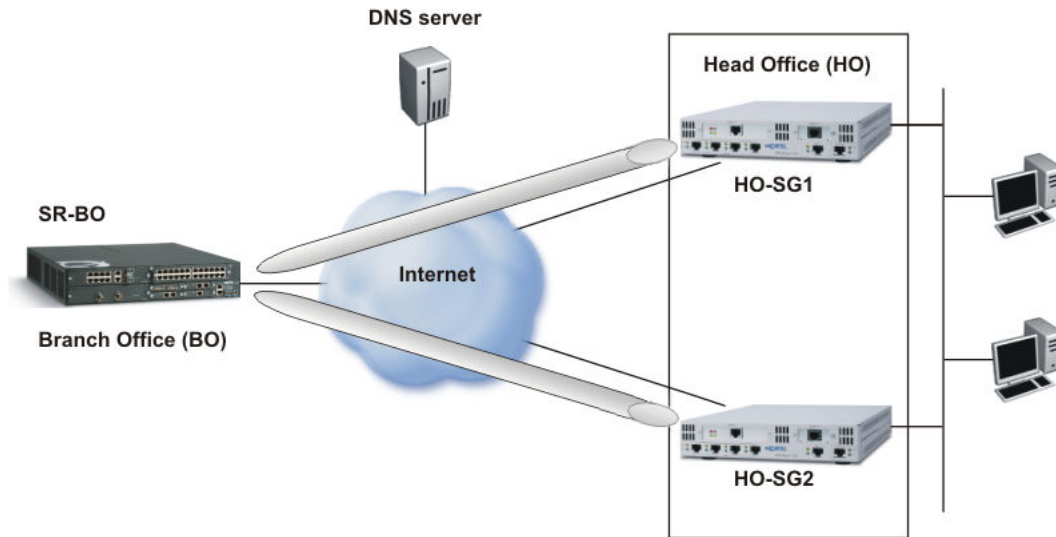
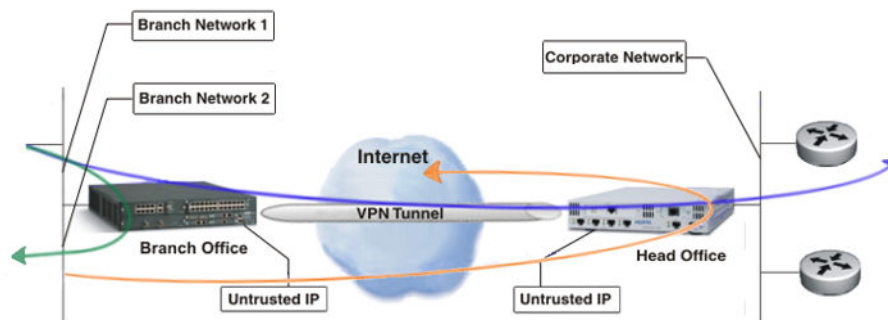


Figure 16: Tunnel failover – round robin DNS

## IPSec VPN bypass policy

The IPsec VPN bypass policy enables system administrators to add exceptions to a generic IPsec policy, thereby selectively choosing the traffic that can or cannot be subjected to VPN processing.

The following diagram illustrates bypass scenarios for a site-to-site VPN between a branch office and a head office network:



The network configuration shown in the diagram above includes the following:

- Bypass traffic between two trusted subnets (depicted using GREEN arrow)
- Bypass traffic from a trusted network to the Branch Office router.

- Bypass Branch Office to internet traffic through the Head Office (depicted using ORANGE arrow).
- Bypass traffic between a pair of subnets at the Branch Office and the Head Office (depicted using BLUE arrow).

**Important:**

An IPSec VPN bypass policy does not affect the function of a firewall policy on the same network.

---

## IPSec nailed up tunnel

With on-demand IPSec connections, a branch office tunnel remains up only when it is transporting traffic. IKE negotiates Phase1 and Phase2 security associations (SAs) between peers (head office and branch office) when traffic matches the selected IPSec policy. However, the initial data (one or more packets) is discarded until the tunnel establishment is complete. After the connection is established, data can be successfully delivered.

In some network scenarios, it can be necessary to have some branch office tunnels remain up, even when there is no traffic traversing the tunnel.

With the IPSec nailed up tunnel feature, branch office connections are formed when a policy is configured and do not require the presence of data to trigger the establishment of a tunnel. Nailed up tunnels are established when the policy is configured on both endpoints. After the tunnel is initially established, when data arrives for a remote network, the data can be delivered with no packet loss.

Nailed up branch office tunnels retry the IKE negotiation with peers until SAs are successfully setup.

Dead Peer Detection (DPD) periodically checks the health status of the nailed up tunnel and flushes the SAs when a peer is found to be unreachable.

Nailed up tunnels are supported on IPSec policies that specify either an IP address object or a specific IP address in the policy. You can specify a maximum of three source ranges and three destination ranges in an address object.

---

## Remote access VPN

Remote access VPN allows individual users such as telecommuters to connect to a corporate network. The user laptop must be equipped with a VPN client that allows a policy to be defined so traffic destined to the corporate network is protected. When the VPN client detects an access point to the corporate network, a secure tunnel to the security gateway (that is, the VPN server) at the corporate headquarters is created.

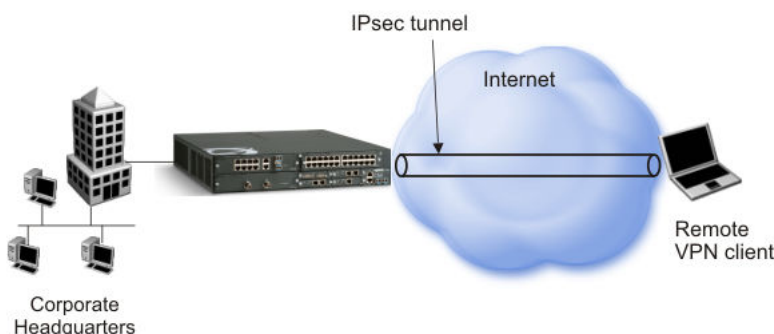
Typically, in this application, the IP address of each VPN client is not known to the VPN server prior to session initiation and therefore cannot be specified in the server configuration.

The VPN client can initiate the tunnel request to the VPN server using IKE main mode or aggressive mode. With main mode for remote access, digital certificates are required. Main mode with pre-shared key is not supported in remote access VPN. Aggressive mode is generally used in remote access VPN solutions. In remote-access applications, it is best to assume that there is a NAT in the middle, which requires NAT traversal support for VPN clients behind devices. The Avaya VPN client and Safenet client support main mode and aggressive mode.

IKE authenticates the VPN server and client. To authenticate the user via a login and password prompt, Mode configuration is used, and optionally Xauth.

The Secure Router 2330/4134 supports Avaya VPN Client version 08\_01.008 (for Windows XP), Avaya VPN Client version 10.01.052 (for Windows Vista), and Apani Client 3.5 (for Linux). All the clients are tested in both aggressive mode and main mode.

You require a minimum of Avaya VPN client version 6.01.146 for Microsoft XP and version 10.01.052 for Microsoft Vista.



**Figure 17: Remote access**

---

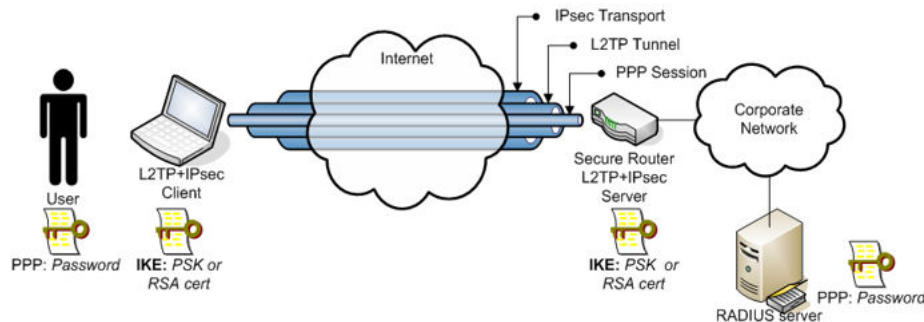
## Remote access VPN with L2TP server

Remote access VPN with L2TP server is another means to allow individual users such as telecommuters to connect to a corporate network. The user's laptop operating system (such as Microsoft Windows or Apple Mac OSX) provides a native VPN/dial-up interface facility which can be configured to use L2TP secured by IPsec.



As shown in the following figure, Internet tunneling is provided by L2TP, which can carry PPP across IP networks. IPsec provides security with Data Privacy, Integrity, and Mutual authentication of the Windows laptop and the Secure Router.

However, with L2TP VPN, IPsec does not provide access control. In this case, you can configure the firewall component of the Secure Router to provide the required access control.



**Figure 18: Remote access VPN with L2TP server**

Each new remote client session becomes a new virtual interface to the Secure Router.

IKE authenticates the VPN server and client, and PPP authenticates the user via a password/login prompt.

User Authentication can be handled by a user list on the Secure Router, or through a RADIUS server. RADIUS servers can interact with Windows Active Directory Domains to authenticate users

The Secure Router 2330/4134 supports the Native Windows L2TP/IPsec client for remote access VPN with L2TP server.

## Supported IPsec security protocols

The Secure Router 2330/4134 supports ESP (Encapsulating Security Payload) and AH (Authentication Header) security protocols. Also, IPsec supports AH over ESP (Security Association Bundle) where ESP is applied on the packet and AH is applied on top of it.

ESP provides the following protection:

- Confidentiality
- Data integrity
- Access control
- Anti replay protection

AH provides the following protection:

- Data integrity
- Access control
- Anti replay protection

---

## IPsec modes

IPsec supports Tunnel mode and Transport mode.

Tunnel mode is used to create VPNs where an entire IP packet is secured and encapsulated into another IP packet along with the required security protocol information.

Transport mode is used when protection is required for packets that already encapsulated (or tunneled) using other protocols such as GRE and IPIP. For information on GRE and IPIP tunnels, see [GRE and IPIP tunneling fundamentals](#) on page 109.

---

## Shared key negotiation with IKE

An important prerequisite for IPsec is to have an authenticated, symmetric key that is shared between the gateways. Such a key can be established either through the process of negotiation between the gateways or through manual configuration (manual configuration is not supported on the Secure Router 2330/4134.) The Secure Router 2330/4134 supports the Diffie Hellman key exchange using Internet Key Exchange (IKE) protocol for the negotiation of the authenticated symmetric key.

IKE provides the following services for IPsec:

- Negotiation of security parameters between IKE peers
- Authentication of IKE peers (using certificates or pre-shared key)
- Key Generation for encryption and hashing

---

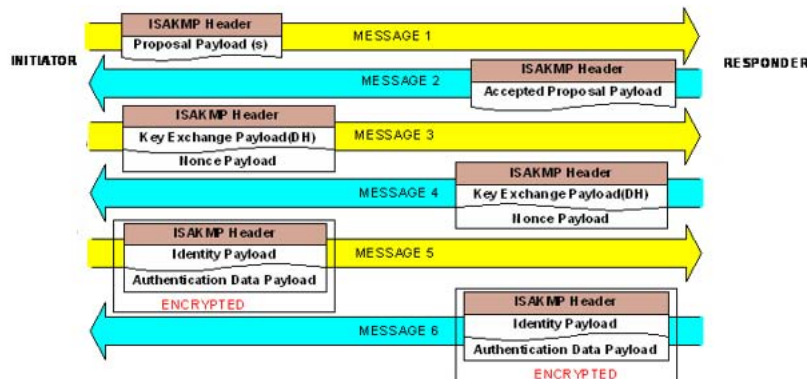
## IKE modes

With IKE, the shared key negotiation is carried out in two phases.

## Phase 1: main mode or aggressive mode

The intent of phase 1 is to establish the authenticated symmetric key and create an IKE security association. This can be achieved using one of two modes: main mode or aggressive mode.

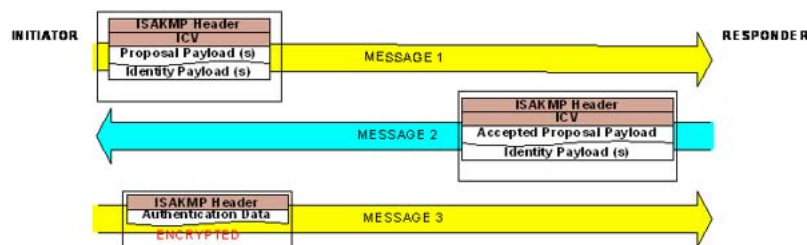
- **Main mode:** Main mode provides for a powerful and flexible negotiation mechanism involving six message exchanges between the security gateways. It also provides identity protection for the parties involved in the negotiation. This is normally used in site to site VPN applications.



**Figure 19: IKE main mode**

- **Aggressive mode:** Aggressive mode provides a quicker negotiation mechanism involving only three message exchanges. However, aggressive mode does not provide identity protection.

Aggressive mode is normally used in remote access VPN applications.



**Figure 20: IKE aggressive mode**

## Phase 2: quick mode

In phase 2 of shared key negotiation, a security association is negotiated over the IKE security association created in phase 1. This is achieved using quick mode. Quick mode provides for negotiating various security parameters between the gateways.

## Key usage extension checking

One of the authentication methods employed in IKE is digital signature based peer authentication. IKE uses a X.509 digital certificate to perform peer authentication. X.509 digital certificates include a key usage extension that defines the purpose of the public key contained in the certificate. Key usage extension checking ensures that a digital certificate contains a specified key usage constraint or the SA establishment with the peer is not authorized.

The purpose of the key usage extension checking feature is to optionally verify whether a peer certificate, issued by a CA, is intended for digitalSignature or nonRepudiation.

Key usage extension checking is supported in main and aggressive modes.

The following table summarizes the logic for peer X.509 digital certificate validation with key usage extension checking.

Key usage extension checking status in the IKE policy	KeyUsage extension status in the peer certificate	Secure Router action
Disabled	Not present	Accepts the certificate from the peer for authentication processing and continues SA establishment.
Disabled	Present	Accepts the certificate from the peer for authentication processing and continues SA establishment
Enabled	Present, but does not mention digitalSignature or nonRepudiation.	Rejects the certificate from the peer for authentication processing, fails the current SA establishment, and logs an appropriate message.
Enabled	Present, and does mention either digitalSignature or nonRepudiation	Accepts the certificate from the peer for authentication processing and continues SA establishment.

Key usage extension checking supports Microsoft CAs.

---

## Peer authentication methods for IKE

The symmetric key derived using Diffie Hellman exchange in IKE is unauthenticated and therefore susceptible to man-in-the-middle attacks. The IKE protocol supports multiple methods to provide authentication of the peers. The Secure Router 2330/4134 provides support for the following authentication methods:

- Pre-shared key

The pre-shared key authentication method relies on a secret key (or password) being shared by the gateways. This password is one of the parameters used by both the gateways in generating a symmetric key after Diffie Hellman exchange. However, for large meshed topologies, it can become onerous to maintain a unique set of passwords for each pair of peers given that, the number of key pairs increases exponentially for each new peer that is added to the topology.

Furthermore, with remote-access configurations, a single pre-shared-key on the Secure Router 2330/4134 is not unique as it must be shared by all remote-access clients. In this case, if one remote client is compromised, then all pre-shared keys must be updated. It is also more difficult to ensure the integrity of a password that is shared by multiple clients.

- RSA or DSS signatures

Signature based authentication provides for an explicit form of peer authentication after the Diffie Hellman exchange. This involves public key cryptography where each gateway provides a digital signature of the negotiation to the other side. Verification of the signature provides authentication of the peers. This method is more secure, but requires a Public Key Infrastructure using X.509 digital certificates. RSA and DSS are two different types of popular digital signatures in use and both are supported in Secure Router 2330/4134 IKE authentication.

For details on the configuration requirements for RSA and DSS signatures, refer to [Digital Certificates in IKE](#) on page 70.

---

## Client configuration and user authentication for remote access VPN

With remote authentication, IKE inherently provides machine level authentication of the VPN server and client. To authenticate the user as well with a login and password prompt, you can optionally configure mode configuration and Xauth.

## Client configuration with mode configuration for remote access VPN

The objective of mode configuration is to make the VPN client appear to be part of the private trusted network (after the packet undergoes decryption in the VPN server).

In order to achieve this, during IKE negotiation, right after phase1 and before phase2, the VPN server can allocate a private IP address to the VPN client. The client then uses this address as the source IP address in the inner IP header.

Optionally, the server can supply DNS and WINS server addresses to the client. The VPN client then installs a virtual IP adapter in its host based on the IP address provided by the VPN server.

A range of IP addresses can be allocated to each IKE policy so that an IP address from this pool can be allocated to the remote users that are configured to use that IKE policy.

Since the address range being allocated to each IKE policy is known, configuring inbound firewall policies is made easier.

This mode-cfg step is also referred to as phase 1.5.

## User Authentication with Xauth for remote access VPN

Xauth is optionally performed right after the modecfg step (phase 1.5) and before phase 2 in IKE exchange. Xauth uses legacy authentication mechanisms such as PAP or CHAP verified against a locally configured username and password database or against a RADIUS server.

---

## Digital Certificates in IKE

Signature based IKE authentication (RSA/DSS) provides an alternative form of peer authentication to PSK. Digital Signature Authentication involves public key cryptography and is a stronger and more scalable solution than PreShared Key authentication. Each gateway provides a digital signature of the negotiation to the other side. Verification of the signature provides authentication of the peers.

In order to perform signature based authentication, each security gateway needs access to the public key of the peer. The public key information is exchanged between the gateways by exchanging digital certificates.

A Public Key Infrastructure (PKI) is required to verify the authenticity of the peers. The PKI assumes that a Certificate Authority that is trusted by both peers is available to create digital certificates.

A Certificate Authority (CA) issues digital certificates conforming to the X.509 format. A digital certificate contains the credentials and public key information of an entity that is endorsed by the CA using a digital signature.

To validate the exchanged certificates, the two security gateways must have a mutually trusted CA.

Each gateway can confirm that the CA validates the identity of the other member. To validate the certificate of the peer, each member checks the certificate revocation list (CRL) issued by the CA. If the peer certificate is not on the CRL, then it is assumed to be valid.

The maximum number of CA certificates supported on the Secure Router 2330/4134 is 10.  
The maximum number of self certificates supported on the Secure Router 2330/4134 is 10.

---

## Internet X.509 PKI certificate and CRL profile

The Secure Router 2330/4134 supports RFC2459, which describes the X.509 v3 certificate format. The RFC also defines the X.509 v2 CRL format and extension set. The Secure Router 2330/4134 only supports the following CRL extensions:

- Key Usage
- Subject Alternative Name
- CRL Distribution Points

---

## X.509 digital certificate compliance with RFC 2253

The X.509 digital certificate format includes subject name and issuer name fields. The subject name identifies the owner of a particular public key or private key pair and the issuer name identifies who certified the subject certificate. X.509 defines the subject name and issuer name as Distinguished Names.

In previous releases, the Secure Router did not support the use of special characters with the Distinguished Name attribute. Beginning with release 10.3, the Secure Router supports RFC 2253, which permits using special characters in the Distinguished Name attribute.

This implementation follows the RFC 2253 to convert an attribute value from ASN.1 to a character string. A UTF-8 string can be used as the string representation of the attribute value if the string does not include any of the following characters, which need escaping:

- a space or "#" character occurring at the beginning of the string
- a space character occurring at the end of the string
- one of the characters ",", "+", ":", "\"", "<", ">" or ";"

If a character to be escaped is one of those listed above, it is prefixed by a backslash ('\ ASCII 92).

The following limitations from releases prior to 10.3 have been eliminated to make the Secure Router compliant with RFC 2253:

1. Server certificate subject DN string gets unstructured name when commas exist within one of the components (i.e. CN part) of the DN string. For example: Cert with

subject dn shown in the application such as: "cn="miao,wu,1234", ou=sust\_sqa1, ou=third\_floor, ou=north\_lab, o=lab, c=us" will become unstructuredName.

2. Server certificate subject DN string gets truncated when backslash and commas exist within one of the components (i.e. CN part) of the DN string. For example: Cert with subject dn shown in the application such as: "CN=miao\,wu\,1234, OU=sust\_sqa1, O=lab, C=us" will become /C=us/O=lab/OU=sust\_sqa1/CN=miao\.
3. Server certificate subject DN string gets truncated when multiple same components (i.e. OU part) separated by commas exist within the DN string. For example: Cert with subject DN shown in the application such as: "CN=miao, OU=sust\_sqa1, OU=third\_floor, OU=north\_lab, O=lab, C=us" will become /C=us/O=lab/OU=sust\_sqa1/CN=miao.
4. Server certificate subject DN string gets truncated when comma exist within one of the components (i.e. O part) of the DN string. For example: Cert with subject dn shown in the application such as: "CN=L.Eagle,O=Sue\, Grabbit and Runn,C=GB" will become /C=GB/O=Sue\CN=L. Eagle.
5. Server certificate subject DN string gets truncated when carriage return character exist within one of the components (i.e. CN part) of the DN string. For example: Cert with subject dn shown in the application such as: "CN=Before \0Dafter,O=Test,C=GB" will become /C=GB/O=Test/CN=Before.
6. Certificates which subject DN with no spaces after commas within one component (i.e. CN part) will not get authenticated.

You must consider the following limitations when configuring special characters for the Distinguished Name attribute:

- The Secure Router does not support an OID in the attribute type to be prefixed by one of the character strings "oid." or "OID.". For example, a certificate with subject DN containing OID: 1.3.6.1.4.1.1466.0=#04024869, O=Test, C=GB.
- The Secure Router does not support multi-valued RDN, the outputs from adjoining AttributeTypeAndValues are separated by a plus ('+' ASCII 43) character. For example, multi-valued RDN: OU=Sales+CN=J.Smith, O=Widget Inc., C=US.
- The Secure Router does not support the replacing the special character to be escaped by a backslash and two hex digits, which form a single byte in the code of the character. For example, SN=Lu\C4\8Di\C4\87.
- Multiple same components separated by commas within the DN string are restricted to three. For example, "CN=miao, OU=sust\_sqa1, OU=third\_floor, OU=north\_lab, O=lab, C=us". The Secure Router does not support allowing multiple same components within the DN string into certificate in manual enrollment mode.
- Remote ID data under an IKE policy should be configured in double quotes ("").
- A country name character string length is restricted to 2 characters.
- The Secure Router does not support X.500 attribute types "STREET" and "UID" in a DN character string.



## Sample configurations

This section provides examples of X.509 digital certificates configured with special characters in the Distinguished Name attribute.

1. To configure a certificate with the subject name "**CN=Charlet, Ricky, O=Avaya, C=US**", enter the following for the trustpoint subject name:  
`"CN=Charlet\, Ricky, O=Avaya, C=US"`
2. To configure a certificate with the subject name "**CN=miao,wu,1234, O=Avaya, C=US**", enter the following for the trustpoint subject name:  
`"CN="miao,wu,1234", O=Avaya, C=US"`
3. To configure a certificate with the subject name "**CN=L.Eagle,O=Sue, Grabbit and Runn,C=GB**", enter the following for the trustpoint subject name:  
`"CN=L.Eagle,O=Sue\, Grabbit and Runn,C=GB"`
4. To configure a certificate with the subject name "**CN=Before0Dafter,O=Test,C=GB**", enter the following for the trustpoint subject name:  
`"CN=Before\0Dafter,O=Test,C=GB"`

---

## Certificate validation

With PKI, the security gateways need to verify the validity of the digital certificates exchanged during IKE negotiation. A certificate is revocable by the CA for a variety of reasons, for example, at the request of a user if the private key is compromised.

To confirm the validity of the certificate, the CA periodically publishes a certificate revocation list (CRL) which contains the list of serial numbers of the revoked certificates.

The Secure Router 2330/4134 uses LDAP to download the CRL periodically from the CA and store it locally so that the validity of certificates can be verified during negotiation. The Secure Router 2330/4134 also supports certificate validation using OCSP.

## Certificate validation using OCSP

Online Certificate Status Protocol (OCSP) is an alternative to the CRL approach for verifying the validity of a digital certificate.

In the CRL method, there can be a lag between the CA publishing an updated CRL and the Secure Router 2330/4134 downloading the same CRL, creating a window of vulnerability. In this period, the router does not have a fool proof mechanism for validating the certificate. OCSP is used to overcome this vulnerability.

In the OCSP method, for each certificate that the router receives, the gateway proactively contacts the CA or the approved third party and requests the status of the certificate in question.

Based on the response of the provider, the gateway makes a decision to accept the certificate or not.

The Secure Router 2330/4134 runs an OCSP client and the CA or an approved third party runs the OCSP server. OCSP uses HTTP for transport.

---

## Certificate enrollment using SCEP client

Since the secure router needs to present a mutually-trusted certificate to a peer, it needs access to the self certificate (the certificate of the secure router itself), the certificate of the issuing authority and the certificate of every CA above that up to the root CA.

The Secure Router 2330/4134 can obtain the CA certificate online, or enroll and import a certificate using Simple Certificate Enrollment Protocol (SCEP).

The secure router supports the SCEP client while the CA runs a SCEP server. SCEP uses HTTP for transport.

The Secure Router 2330/4134 obtains a certificate using SCEP as follows:

- The router creates a public/private key pair.
- The router sends the public key to the CA, requesting the CA certificate from the CA.
- The router generates a request for a self certificate and submits the request to the CA.
- The CA verifies the certificate request and sends a signed certificate to the router.
- The router stores the CA certificate and self certificate in the PKI database for future use.

The secure router can use the certificates in IKE to establish IPsec security associations between two gateways.

---

## Manual certificate enrollment

As an alternative to SCEP, the Secure Router 2330/4134 also supports manual certificate enrollment. The steps that you must follow for manual enrollment are as follows:

- Manually upload the CA certificate (using cut-and-paste)
- Generate self certificate request
- Manually submit the self certificate request to the CA (using cut-and-paste)
- Manually upload the approved self certificate from the CA (using cut-and-paste)

## RSA certificate key size

Beginning with software release 10.3, the maximum key size for RSA certificates is increased to 4096 bits. The Secure Router now supports RSA keys with 512, 1024, 2048, 3072, and 4096 bits. The factory default is 1024 bits.

The following certificate authorities (CAs) support RSA key sizes of 3072 and 4096 bits:

- OpenSSL
- Microsoft 2003/2008 Server

### Limitations

Key pair generation, signature, and verification for certificates with 3072 and 4096 bit keys are not hardware assisted on the Secure Router, which can result in delays when a large key is used. The slower CPU speed of the SR2330 can result in longer such delays than the SR4134. The following table lists the expected average certificate download wait times related to key size for the Secure Router 4134 and 2330.

Secure Router	Key Size (bits)	Expected wait time
<b>4134</b>	512	<30 seconds
	1024	<30 seconds
	2048	<30 seconds
	3072	<30 seconds
	4096	45 seconds
<b>2330</b>	512	30 seconds
	1024	1 minute
	2048	2 minutes
	3072	3 minutes
	4096	4 minutes 30 seconds

Because Diffie-Hellman group key generation performs certificate signature and certificate verification during IKE negotiation, the IKE tunnel setup rate deteriorates when you use 3072 and 4096 bit RSA keys to establish a tunnel between the Secure Router and the branch office. This performance degradation is especially pronounced when you attempt to establish a large number of branch tunnels simultaneously.

When you use a 3072 or 4096 bit RSA key and enter the `crypto ca enroll` command, the command line can become unresponsive until the key generation is complete. A system message indicating the wait time is displayed.

---

## Dead peer detection

The Secure Router 2330/4134 provides support to detect when an IKE peer gateway dies unexpectedly. This prevents a situation whereby packets are tunneled to a black hole, resulting in bandwidth loss and recovery problems. The Secure Router 2330/4134 supports RFC3706, which describes a method, called Dead Peer Detection (DPD), to confirm the status of peer gateways.

### On-demand DPD

With on-demand DPD, the Secure Router sends a query for the operational status of an IKE peer when the router has traffic to send to the peer, but has not received traffic from the peer within a predetermined time interval (transmit interval). The router then sends a series of keepalive messages at a predetermined time interval (retry interval). If there is no response to this series of keepalive messages, the Secure Router determines that the peer is in a dead state, and deletes the IPsec and IKE SAs to the peer.

### Periodic DPD

Periodic DPD provides the Secure Router with stateless failover by querying the operational status of IKE peers at a regular, predetermined time interval (transmit interval). If the Secure Router does not receive a response to a status query from an IKE peer, the router then sends a series of keepalive messages at a predetermined time interval (retry interval). If there is no response to this series of keepalive messages, the Secure Router determines that the peer is in a dead state, and deletes the IPsec and IKE SAs to the peer.

Periodic DPD offers earlier dead peer detection than on-demand DPD, but periodic DPD relies on frequent queries between the Secure Router and the IKE peer, which results in a higher volume of network traffic.

When the Secure Router is required to communicate with a large number of IKE peers, you are advised to use on-demand DPD.

If you do not configure the keepalive mode for periodic DPD, the Secure Router defaults to on-demand DPD.

---

## Nat Traversal support

During IKE negotiation, the Secure Router 2330/4134 automatically detects NAT in the middle between two security gateways. Since NAT in the middle can affect the integrity of the secure packets (ESP or AH), upon NAT detection, the Secure Router 2330/4134 automatically uses NAT traversal protocol. This protocol provides an additional UDP encapsulation over the

secure packets. This is applied to all subsequent IKE negotiations as well as to the secure packets.

When a NVR device serves as an IKE peer to a Secure Router, the Secure Router must be configured to use RFC 3947 for NATT.

For NAT traversal acceptance configuration information, see [Configuring RFC 3947 NAT traversal acceptance](#) on page 145.

---

## Larger DH groups for branch office tunnels

The strength of Diffie-Hellman (DH ) and RSA algorithms are approximately equal for the same key size. Beginning with software release 10.3, two additional DH groups (14 and 15) are available with the Secure Router, to maintain consistency with the increased RSA key size, and provide additional options for IKE negotiations for branch office tunnels.

---

## Multiple IKE proposals

IKE establishes a secure communication channel for itself in phase 1 before negotiating the IPsec proposals in phase 2. During Phase 1, IKE can propose up to five protection suites. Each IKE proposal specifies a particular choice for the following:

- authentication method
- encryption algorithm
- hash algorithm
- Diffie-Hellman (DH) group
- lifetime

At least one proposal in the list must be agreeable to both peers for the negotiation to proceed. Only one proposal on the list is ultimately negotiated and used by the peers.

The Secure Router 2330/4134 supports a comprehensive and flexible protection suite to converge with several peers with dissimilar security capabilities. The following table describes the security elements supported by the Secure Router 2330/4134 in phase 1 IKE negotiation.

**Table 6: Supported elements in phase 1 IKE**

Security element	Values	Comments
Authentication Method	PSK, RSA-SIG, DSS-SIG	Signature Authentication is Hardware Accelerated

Security element	Values	Comments
Encryption algorithms	DES, 3DES, AES-128, AES-192, AES-256	
Hash algorithms	MD5, SHA1	
DH group	Group1, Group2, Group5, Group14, Group15	Used for deriving IKE Phase 1 key material. DH exponentiation is hardware accelerated.
Security association lifetime	Time (in seconds) or volume of traffic (in kilobytes)	
Number of proposals per policy	Five	Provides flexibility in negotiation

## Multiple IPsec proposals

After IKE establishes a secure communication channel for itself in phase 1, it proceeds to negotiate the IPsec proposals in phase 2. During Phase 2 IKE may propose multiple Protection Suites for IPsec protocols such as ESP and AH. Each phase 2 proposal specifies a choice for the following:

- encryption algorithm
- hash algorithm
- lifetime
- encapsulation mode

Phase 2 proposals can specify a list of AND proposals for ESP and AH. The phase 2 proposals can also specify a list of OR proposals for ESP with proposal choice set 1 or ESP with proposal choice set 2. The following example illustrates some of the possibilities:

1. ESP AND AH with 3DES, SHA1, 2000 seconds, tunnel mode
2. OR ESP AND AH with DES, MD5, 2000 seconds, tunnel mode
3. OR ESP with AES128, SHA1, 2000 seconds, tunnel mode
4. OR ESP with AES128, SHA1, 1000 seconds, tunnel mode
5. OR AH with SHA1, 1000 seconds, tunnel mode

At least one proposal in the list must be agreeable to both peers for the negotiation to proceed. Multiple phase 2 proposals can be negotiated, one for each protocol (ESP and AH).

The Secure Router 2330/4134 supports a comprehensive and flexible protection suite to converge with several peers with dissimilar security capabilities. The following table describes the supported security elements in phase 2 IKE negotiation.

**Table 7: Supported elements in phase 2 IKE**

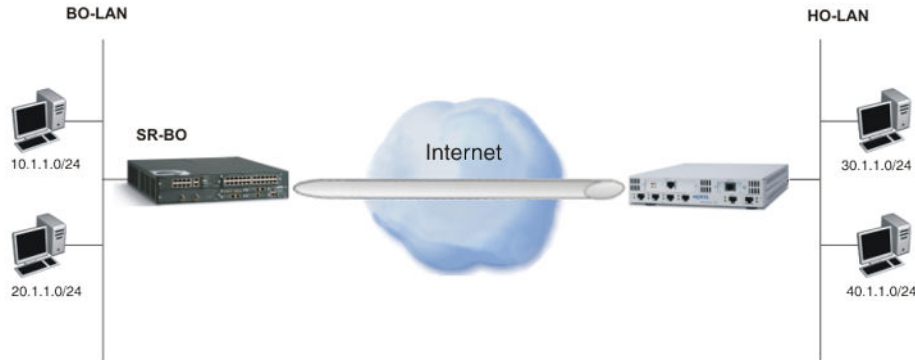
Security element	Values	Comments
Encryption algorithms	DES, 3DES, AES-128, AES-192, AES-256, null	Hardware accelerated
Hash algorithms	MD5, SHA1, null	Hardware accelerated
PFS group	Group1, Group2, Group5	Used for deriving IPsec key material.
Security association lifetime	Time (in seconds) or volume of traffic (in kilobytes)	
Number of proposals per policy	Five	Provides flexibility in negotiation

---

## Multiple networks in a single IPsec policy

The Secure Router IPsec policy supports specifying multiple ranges of source and destination network addresses as IP address objects. The IP address objects are variables that you can assign to a single IPsec policy. This feature avoids the necessity of configuring multiple IPsec policies when multiple source and destination networks need to be secured on the same gateway.

For example, in the following topology diagram, two ranges of branch office LAN (BO-LAN) source IP addresses (10.1.1.0 to 10.1.1.255 and 20.1.1.0 to 20.1.1.255) are specified as a common source IP address object, and two ranges of head office LAN (HO-LAN) destination IP addresses (30.1.1.0 to 30.1.1.255 and 40.1.1.0 to 40.1.1.255) are specified as a common destination IP address object. The source and destination IP address objects are assigned to a single IPsec policy.



**Figure 21: Multiple networks in a single IPsec policy**

When traffic matching the policy selectors is received, IKE negotiates Phase2 SAs for the address combination in the IPsec policy. A pair of phase2 SAs are created for each combination source and destination network address.

When DPD detects that a peer is unreachable, the Secure Router flushes all the Phase2 SAs associated with the peer.

You can configure IP address objects in either an IP address subnet or IP address range format. If you configure an IP address object in the IP address range format, you can add multiple network addresses to the object. If you configure an IP address object in the IP address subnet format, you can add only one network address to the object.

The Secure Router supports multiple networks in a single IPsec policy only when IP address objects are configured in the IP address range format.

---

## Identifying traffic to be encrypted with VPN

The IPsec VPN tunnel carries protected traffic from one trusted network to another trusted network. The Secure Router 2330/4134 provides the **crypto** command to configure policies and parameters for IKE and IPsec for the creation of VPNs.

To create the VPN, you must identify one untrusted interface that serves as the local endpoint for the creation of the VPN tunnel (using the **crypto untrusted** command).

To identify the IP stream requiring encryption, you must use the **match** command. This command allows you to specify at minimum the source IP address (or range) and destination IP address (or range) of the protected stream. Additional filter options are also available to specify a more granular stream.

However, defining the match policy alone is not sufficient to identify the tunnel traffic. To ensure that traffic identified in the match command is forwarded through the tunnel, you must identify at least one crypto trusted interface that specifies where the source traffic enters the router



(using the `crypto trusted` command). This command identifies the interface as the source of tunnel traffic.

If traffic meeting the `match` filter rules enters the router through an interface that is not identified as `crypto trusted`, the traffic is not encrypted using the VPN.

---

## Prioritizing IPsec policies

You can prioritize IPsec policies by specifying the order in which policies are applied.

When you create a new IPsec policy, you can identify the existing policy after which the new policy is inserted (after-name), and the existing policy before which the new policy is inserted (before-name).

If you create an IPsec policy without specifying a before name and an after name, the new policy is appended at the end of the list of existing IPsec policies.

You cannot assign both an after name and before name to the same IPsec policy.

You cannot assign an after name or before name to a previously existing IPsec policy.

When you remove an IPsec policy, you do not have to specify an after name or before name.

---

## Firewall considerations for trusted and untrusted VPN interfaces

Identifying an interface as `crypto trusted` or `crypto untrusted` has implications for the firewall treatment of that interface.

When you set the external interface as `crypto untrusted`, the interface is automatically added to the internet untrusted firewall zone.

For the untrusted `crypto` interfaces, you need to configure a firewall policy to allow IKE negotiations to the local tunnel interface.

When you set the internal interface as `crypto trusted`, the interface is automatically added to the trusted corp firewall zone. If you want the trusted interface to belong to a different trusted firewall zone, you must configure this preference using firewall commands.

For the trusted `crypto` interfaces, to allow VPN tunnel traffic to pass through the firewall from the untrusted interface, you must configure an inbound policy on the trusted interface.

By default, most trusted outbound traffic usually meets the default policy of `1024 out allow` (unless this default policy has been administratively altered), and so this traffic will pass.

---

## Routing considerations for VPN (and firewall)

The security processing must be aware of the route not only to the destination network (which is always required for basic routing), but also the route to the source network. Knowledge of the route to the source network is used by the firewall to prevent spoofing.

---

## Perfect forward secrecy

The Secure Router 2330/4134 supports the following two types of perfect forward secrecy (PFS).

- key associated PFS
- identity associated PFS

Typically the DH key agreement in IKE phase 1 is used to generate keying information for both phase 1 and phase 2 keys. With key associated PFS, the Secure Router 2330/4134 performs a second, optional DH key agreement in IKE phase 2. The optional DH agreement ensures that the keying information for IPsec SAs is unrelated to the keying information for IKE SAs. This is a trade off of increased security for extra computational effort during session establishment. Enabling key associated PFS reduces the Secure Router 2330/4134 maximum tunnel establishment rate by one half. You can configure PFS for keys with the `crypto ipsec pfs-group <group>` command.

With identity associated PFS, the Secure Router 2330/4134 drops the IKE SA immediately after the IPsec SA is established. When a new IPsec SA is required (for example, during re-key) a new IKE SA must be established. The establishing of a new IKE SA protects the ID information of the initiator with a fresh DH exchange for every session establishment. Establishing a new IKE SA has no benefit in common scenarios where the IKE and IPsec lifetimes are the same, and new IKE SAs appear at approximately the same frequency as new IPsec SAs. This also has no benefit when you use IKE aggressive mode, since with IKE aggressive mode, the ID information of the initiator is already protected. Immediately dropping the IKE SA leaves the two VPN gateways without a control channel between them for exchanging notification information. Identity associated PFS interoperates very poorly with non-Secure Router peers. You can configure identity associated PFS with the `crypto ike pfs` command.

---

## Security Policy Database

The SPD determines which traffic going through the gateway needs to be secured. In order to determine this, you must configure policies and specify the required level of security. You can

specify the source and destination IP address (host or subnet or range of address) that need to be secured. Optionally, the protocol, source and destination ports can also be configured.

Policies can be administratively enabled or disabled.

---

## PMTU support

Path MTU is a technique used by the hosts to discover the lowest MTU value along the tunnel path. The discovered value can be used to limit the IP packet size to this value so that fragmentation along the path can be avoided.

To support PMTU, the Secure Router 2330/4134 performs the following:

- generates an ICMP message back to the host behind the gateway if the host forwards a packet that needs IP fragmentation but has the Don't Fragment (DF) bit set.
- records the PMTU value in the ICMP message received from the peer security gateway in the corresponding SA and uses the new value for future packets from the corresponding host behind the gateway.
- preserves the DF bit while encapsulating the original packet.

---

## TCP MSS Clamping

The TCP MSS feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse the interface. The `ip tcp-mss` command under the interface section specifies the MSS value on the intermediate router of the TCP SYN packets to avoid truncation. When a TCP SYN packet traverses the router the MSS option field in the packet is lowered to the value specified in the `tcp-mss-clamping` config.

The ability to set the TCP MSS value is supported on Ethernets, Bundles, AVC, L3-VLAN, Tunnels and firewall policy. Firewall policy level MSS clamping gives greater granularity by allowing the clamping to be performed only for certain hosts. When setting the TCP MSS value it is recommended that the MSS value is at least 40 bytes less than the MTU of the interface. The TCP header takes up 20 bytes of data (or more if options are used); the IP header also uses 20 or more bytes. This means that between them a minimum of 40 bytes are needed for headers, all of which is non-data "overhead".

### Important:

TCP MSS Clamping will not work on an Ethernet switch port, if the TCP connection originates from a switched Ethernet port on the router and its TCP destination is over another Ethernet switch port.

---

## Firewall considerations with VPN

To allow VPN connections, you must configure an inbound firewall policy in the internet zone for IKE negotiation that allows self connections to UDP port 500.

In addition, in cases where there is a NAT in the middle between the VPN peers, you must allow IKE self connections to UDP port 4500 to support NAT traversal. In cases where the NAT is enabled on one of the peers themselves, you do not need to open port 4500.

For site to site VPN, it is rare (though possible) to encounter a NAT in the middle between the peers. However, for remote access VPN, remote access clients are most often behind a NAT, in which case, UDP port 4500 must be opened.

Finally, to support L2TP and IPsec remote access VPN, in addition to allowing the IKE port connection, you must configure an inbound self firewall policy in the internet zone that allows L2TP connections to UDP port 1701.

---

## IPSec VPN support without firewall

In previous releases, to enable IPSec VPN you had to enable the firewall. Traffic requiring IPSec services had to undergo firewall-related checks, such as including firewall policy lookup and max-connection limit.

To allow VPN to operate without the firewall, you can now choose to globally disable the firewall on the router.

When you disable the firewall, all traffic skips firewall-related checks. IPSec services are provided based on the policies configured in the security policy database.

Using VPN without firewall is advantageous when designing network VPN failover scenarios. The firewall does not trust currently established TCP sessions appearing on new interfaces and inhibits failover and recovery.

To enable IPSec VPN support without firewall, use the system security **firewall-disable** command.

This feature requires the router to be rebooted to take effect. Also, if there is any firewall related configuration in the configuration file (system.cfg), it is overwritten and lost.

For information about configuring VPN support without a firewall enabled, see [Configuring IPSec VPN Support without Firewall](#) on page 245.

---

## QoS over VPN

Enabling VPN on WAN interfaces complicates the configuration of Quality of Service (QoS) in the WAN outbound direction. Because the packets coming from the Ethernet side are encapsulated and encrypted, the original IP addresses and ports are inaccessible for QoS on outbound WAN traffic.

However, the Secure Router 2330/4134 can support QoS over VPN using DSCP (DiffServe Code Point). While encapsulating (and encrypting) the packet, the secure router preserves the DSCP information by copying the TOS byte information from the inner IP header to the outer IP header. Since the DSCP information is available in the outer IP header also, QoS on the WAN outbound direction can be applied. To enable this configuration, packets arriving at the ingress point on the Ethernet interface must be marked with DSCP.

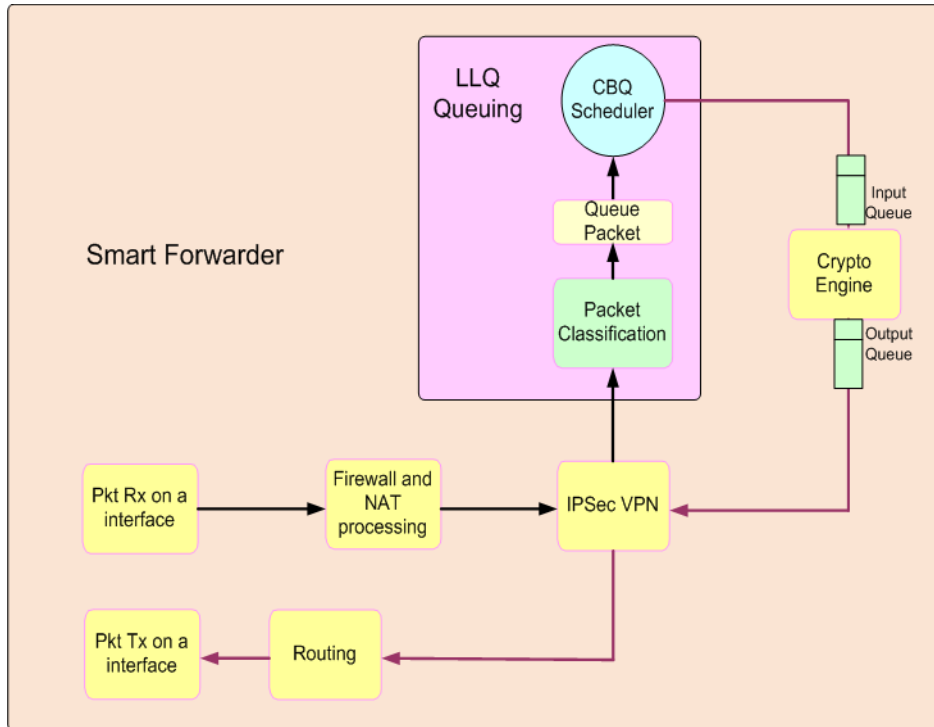
For more information on QoS configuration, see *Avaya Secure Router 2330/4134 Configuration – Traffic Management* (NN47263-601).

---

## Crypto QoS (CBQ) for IPsec VPN

The Secure Router 2330/4134 crypto engine performs encryption and hashing of packets for IPsec VPN tunnels. However, the crypto engine throughput is less than the system throughput, and therefore, there is potential for congestion to build up and packets to be dropped at the crypto engine queue. You can apply CBQ to packets entering the encryption engine to guarantee bandwidth and reduce the latency for delay sensitive voice packets.

As shown in the following figure, the crypto engine maintains an input FIFO queue and an output FIFO queue. The packets needing crypto services (encryption, decryption, hashing, and so on) have to be queued into the input FIFO queue for the crypto engine to act on it. When the processing is completed, the packet is placed on the crypto output queue.



**Figure 22: LLQ**

To avoid congestion on the crypto engine, you can classify traffic into the desired classes using the QoS multifield classifier and apply CBQ with associated committed rate and priority parameters for each crypto class.

Crypto interfaces do not support RED or policing, although marking can be configured.

Packets that are classified into a leaf traffic class are placed in the associated class queue. The CBQ scheduler periodically services the queues of all leaf traffic classes on each crypto interface. The service that each class queue receives depends on the service parameters (CR, PR, Priority) assigned to it. Eight priority levels are supported, from priority 1 (highest) to priority 8 (lowest).

The crypto engine bandwidth varies for different packet sizes. Therefore, for each class you must configure the committed rate as a percentage of the crypto interface bandwidth. The average bandwidth for the interface is calculated by the number of bytes processed by the crypto engine in a given interval. This bandwidth is used to calculate the average interface bandwidth using the exponentially weighted moving average. This ensures low latency without sacrificing the interface bandwidth.

For more information on QoS configuration, see *Avaya Secure Router 2330/4134 Configuration – Traffic Management* (NN47263-601).

---

## Logging and Statistics

VPN provides logging support on a global level, this logging can be on a system console or telnet session of a syslog server. Statistics are maintained for the number of packets and bytes processed in the inbound and outbound direction for each SA. This helps administrator to debug problems, if any.

---

## Standards compliance

The Secure Router 2330/4134 implementation of IPsec VPN complies with the following RFCs:

**Table 8: IKE standards**

RFC number	Description
RFC1191	Path MTU Discovery
RFC1829	ESP DES-CBC Transform
RFC1851	ESP Triple DES Transform
RFC2104	HMAC: Keyed-Hashing for Message Authentication
RFC 2401	Security Architecture for the Internet Protocol
RFC2402	IP Authentication Header
RFC2403	The Use of HMAC-MD5-96 within ESP and AH
RFC2404	The Use of HMAC-SHA-1-96 within ESP and AH
RFC2405	The ESP DES-CBC Cipher Algorithm With Explicit IV RFC 2406 IP Encapsulating Security Payload
RFC2407	The Internet IP Security Domain of Interpretation (DOI) for ISAKMP
RFC2408	Internet Security Association and Key Management Protocol (ISAKMP)
RFC2409	Internet Key Exchange (IKE)
RFC2410	The NULL Encryption Algorithm and Its Use With IPsec
RFC3174	US Secure Hash Algorithm 1 (SHA1)
RFC3526	More Modular Exponential (MODP) Diffie-Hellman groups for IKE
RFC3602	The AES-CBC Cipher Algorithm and Its Use with IPsec

RFC number	Description
RFC3706	A Traffic-Based Method of Detecting Dead IKE Peers
RFC3715	Network Address Translation (NAT) Compatibility Requirements
RFC1853	IP in IP tunneling
RFC2784	Generic Routing Encapsulation (GRE)
RFC2890	Key and Sequence Number Extensions to GRE
draft-ietf--nat-t-ike-01.txt	Negotiation of NAT-Traversal in the IKE
draft-ietf--udp-encaps-01.txt	UDP Encapsulation of ESP Packets
draft-ietf--nat-t-ike-02.txt	Negotiation of NAT-Traversal in the IKE
draft-ietf--udp-encaps-02.txt	UDP Encapsulation of ESP Packets
draft-ietf--nat-t-ike-05.txt	Negotiation of NAT-Traversal in the IKE
draft-ietf--udp-encaps-05.txt	UDP Encapsulation of ESP Packets
draft-ietf--isakmp-xauth-03.txt	Extended Authentication Within ISAKMP/Oakley
draft-dukes-ike-mode-cfg-01.txt	ModeCfg server using The ISAKMP Configuration Method

**Table 9: PKI standards**

RFC number	Description
PKCS #1	RSA Cryptography Standard
PKCS #3	Diffie-Hellman Key Agreement Standard
PKCS #7	Cryptographic Message Syntax Standard
PKCS #10	Certification Request Syntax Standard
RFC2511	Internet X.509 Certificate Request Message Format
RFC2560	Internet X.509 Public Key Infrastructure Online Certificate Status Protocol
RFC3494	Lightweight Directory Access Protocol version 2 (LDAPv2)

**Table 10: L2TP standards**

RFC number	description
RFC2661	Layer 2 Tunnel Protocol (L2TP)
RFC3193	Securing L2TP using IPsec



# Chapter 6: Avaya VPN client fundamentals

Avaya virtual private network (VPN) is an existing Avaya brand of IPsec-based remote access client. Avaya VPN Client is multi-platform (Windows, MacOS, Solaris, HP-UX), simple to configure, and can be highly controlled by the VPN Remote Access Gateway. These advantages are a result of proprietary extensions to Internet Key Exchange (IKE), ModeCFG and Xauth made by Avaya.

The Avaya Secure Router 2330/4134 can serve as a small to medium capacity Avaya VPN server. For a high capacity Avaya VPN server solution, see the Avaya VPN Gateway.

This chapter describes the Avaya VPN Internet Remote Access Server (IRAS) in the Secure Router 2330/4134.

The Secure Router 2330/4134 offers both Site to Site and Remote Access VPN service. With Remote Access service, the Secure Router interoperates with Safenet clients, L2TP+IPsec clients, and with Contivity style clients. For Contivity style clients, the Secure Router 2330/4134 was tested with Avaya VPN Clients versions 6, 8, and 10.

You require a minimum of Avaya VPN client version 6.01.146 for Microsoft XP and version 10.01.052 for Microsoft Vista.

---

## Overview of Avaya VPN client operation

Remote and traveling users can connect to corporate resources by using an Avaya VPN client on a PC. The Secure Router implements the Avaya VPN Internet Remote Access Server (IRAS). The user PC and Avaya VPN client can be anywhere on the Internet, including specifically behind a NAT. The Avaya VPN client (PC) initiates a connection to the Avaya VPN IRAS on the Secure Router 2330/4134 and after successful negotiation, gains a virtual adaptor with an IP address from the corporate network and a secure channel across the Internet to the corporate network. The following diagram shows an example of how a remote or travelling user can use the Secure Router 2330/4134 and Avaya VPN client to connect to a corporate network.

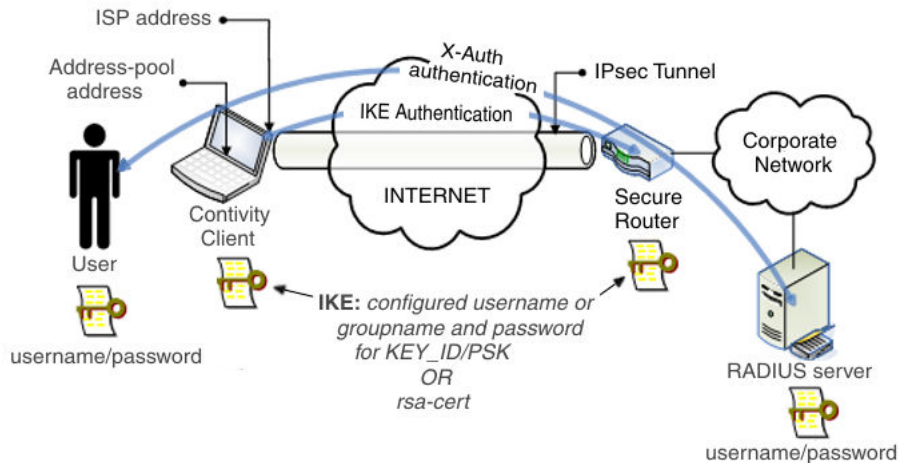


Figure 23: Remote user access to corporate network

## Remote management using Avaya VPN client

Remote and traveling users can also connect to the Secure Router itself for the purpose of managing the Secure Router by using an Avaya VPN client on their PC. The user PC and Avaya VPN client may be anywhere in the internet, including specifically behind a Network Address Translation (NAT). The Avaya VPN client (PC) initiates a connection to the Avaya VPN IRAS on the Secure Router 2330/4134 and after successful negotiation, gains a virtual adaptor with an IP address from the corporate network and a secure channel across the Internet to the Secure Router. The following diagram shows an example of how a remote or travelling user can connect to and manage the Secure Router 2330/4134 with Avaya VPN client.

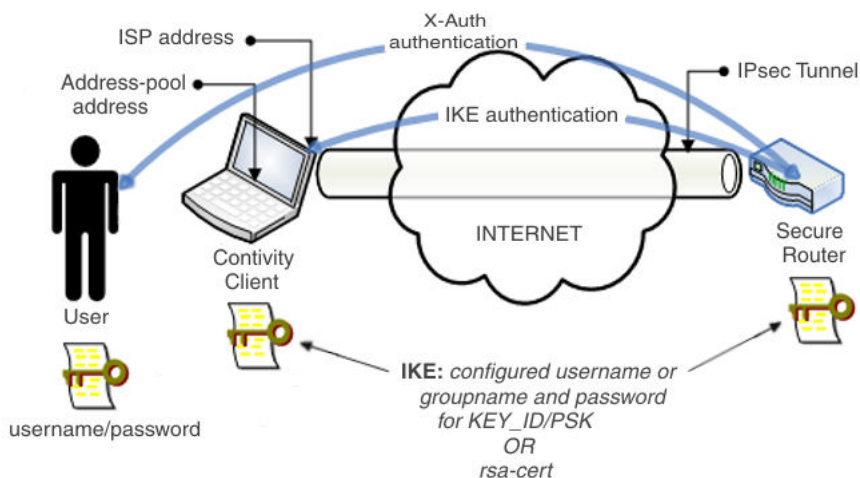


Figure 24: Remote user access to Secure Router for management

## Supported Avaya VPN features

In the current release, the Secure Router 2330/4134 does not support all Avaya VPN features. If you want to upgrade to a Secure Router 2330/4134 for Avaya VPN requirements, review the list of supported features carefully to assess whether the Secure Router 2330/4134 Release 10.3 is sufficient for your application.

**Table 11: Current supported features**

Feature	Description
Client version recognition	The ability to accept the non-standardized client version attribute from an Avaya VPN client.
IKE KEY_ID	The standardized IKE Identification payload property matching a configured string to a key
Auto IKE key generation	The Avaya proprietary ability to generate both a KEY_ID string and a Preshared Key for use in IKE from a username/password or a groupname/password
Authentication methods	Support for the Avaya VPN authentication notions of (1) user authentication, (2) group authentication, and (3) certificate authentication
Proprietary IKE keepalive	Avaya proprietary keepalive
Proprietary NAT traversal	Avaya proprietary auto NAT discovery, encapsulation, and port selection methods
Client configuration push	The Secure Router 2330/4134 IRAS can be configured to provide configuration parameters to Avaya VPN clients
Extra IKE status codes	A few proprietary notify messages

## Client version recognition

Avaya VPN clients send a proprietary IKE SA payload attribute to indicate their version. The Secure Router 2330/4134 recognizes this proprietary attribute rather than failing a non-standardized connection request. The Secure Router 2330/4134 handles this connection as an Avaya VPN connection, which receives slightly different behavior than a standardized ModeCFG/Xauth connection. Since the IKE SA payload is the first payload sent, the Secure Router 2330/4134 can distinguish an Avaya VPN connection from standardized connections.

In the Secure Router 2330/4134 command line interface (CLI), configure Avaya VPN connections under the `crypto contivity-iras` tree rather than the `crypto dynamic` tree which is still used to configure standardized ModeCFG/Xauth client connections.

---

## Authentication

The Secure Router 2330/4134 can interoperate with Avaya VPN clients in the following three authentication methods:

- username and password
- group security
- digital certificate

### Username and password

When Avaya VPN clients are in username and password authentication mode, you must configure each client with a unique username and password pair. On the Secure Router 2330/4134, you must configure an IKE policy with a remote ID of type user. To support multiple clients, you must enter the `remote-id` command repeatedly to specify the matching username and password for each client.

While in username and password mode, these credentials are sufficient and the Avaya VPN client can connect to the Secure Router 2330/4134 with no user interruption.

While in username and password, both the Avaya VPN client and Secure Router 2330/4134 use the username and password as inputs to auto-derive the IKE credentials of KEY\_ID and PSK.

Username and password mode is useful for site administrators who wish to manage a list of credentials on the Secure Router 2330/4134. No other devices are involved in the authentication.

### Group security

To support an Avaya VPN client that is in group ID and password authentication mode, you must configure a corresponding IKE policy on the Secure Router 2330/4134 with a matching remote ID of type group that specifies the group ID and password. You must configure each Avaya VPN client in this group with the same group ID and password.

In group ID and password mode, after IKE phase 1 is complete, the Secure Router 2330/4134 prompts the Avaya VPN client user with a follow-up user name and password. To perform this supplementary authentication, the Secure Router 2330/4134 automatically uses Xauth. No other configuration is necessary to enable Xauth in this mode, and Xauth is not utilized in other Avaya VPN client authentication methods.

However, to support Xauth authentication, you must configure the RADIUS server under the `aaa` radius tree to verify credential requests.

While in group ID and password mode, both the Avaya VPN client and Secure Router 2330/4134 use the username and password as inputs to automatically derive the IKE credentials of KEY\_ID and PSK.

Note that group ID and password mode involves trading off unique IKE credentials for each connection for a group secret. If you assume that a group secret is no real secret at all, then you realize that the entire security of this connection depends upon the secondary credentials.

Group ID and password mode is useful for site administrators who wish to manage (or already manage) a list of credentials in any database (for example, a Windows NT domain) which can be front-ended by a RADIUS server. Although it is still possible to add usernames and passwords to a local database on the Secure Router 2330/4134, site administrators are typically interested in trading off the security of unique IKE credentials for the convenience of interacting with an out-board database of unique credentials.

Secure ID and Axent are not supported in this release.

### **Group security with RADIUS**

When RADIUS is used with Xauth in group security mode, the RADIUS server can return only the verification status, or can also return configuration information for the client. For more information, see [Standard Client Configuration Push](#) on page 96.

The Secure Router 2330/4134 requests a PAP authorization and passes these attributes to the RADIUS server:

1. User-Name
2. User-Password

The Secure Router 2330/4134 forwards these RADIUS attributes from a RADIUS server onward to the Avaya VPN client:

1. Framed-IP-Address
2. Primary-DNS
3. Secondary-DNS
4. Primary-NBSS
5. Secondary-NBSS

The Secure Router 2330/4134 also sends the following to the Avaya VPN client: Framed-IP-Netmask = 32 bits set

## **Digital certificate**

You can also configure Avaya VPN clients to use digital certificate authentication instead of username and password or group-name and password authentication. For instructions about loading a Certificate Authority certificate, and requesting and loading a device certificate, see

*Avaya VPN Router Configuration – Client, NN46110/306.* In the Secure Router 2330/4134, perform the following:

1. Configure a crypto trustpoint.
2. Authenticate the crypto trustpoint.
3. Enroll with the crypto trustpoint.

Once the common CA certificate and unique device certificate have been loaded into the Avaya VPN client and the Secure Router 2330/4134 respectively, configure the Avaya VPN client for the Digital Certificate authentication option and select your recently loaded device certificate for use with the connection. In the Secure Router 2330/4134, configure the contivity-iras policy with a remote ID of type der-encoded-dn. Also in the Secure Router 2330/4134, set the IKE mode to main and the authentication-method to `rsa-sig`.

## Security considerations related to authentication methods

The group authentication option is the most popular of the three authentication options supported with the Avaya VPN client. This section discusses each authentication option and the security considerations related to each.

### Security considerations: Username and password

With the username authentication option, IKE PSK authentication is used with a KEY\_ID form of ID. Both the PSK and the KEY\_ID are derived from the username and password. The derivation algorithm is kept secret by Avaya. In this form of authentication, there is no Xauth authentication layer. An attacker who does not know the secret derivation algorithm cannot mount attacks on the exchange. An attacker who knows the secret derivation algorithm can mount username and password guessing attacks. These attacks increase in effectiveness if the attacker also has knowledge of a username and password. There is not much any security system can do to stop attackers with knowledge of the credentials of authentic users. The defense against this attack is to educate users to keep their credentials a secret.

### Security considerations: Digital certificate

With the digital certificate authentication option, each client machine is configured with a digital certificate through administrative means. For information about configuring the digital certificate, see *Avaya VPN Router Configuration – Client, NN46110/306*. The client initiates an IKE main mode connection with auth-type set to RSA\_SIG. Any IKE standard name-based ID can be used. There is no Xauth layer for user authentication. This is an unmodified IKE exchange using RSA\_SIG auth and is highly resistant to network based attacks. However, an attacker can steal or acquire a laptop with the installed digital certificates and then establish connections appearing to be the correct laptop. The defense against this attack is for trusted members to report when they notice a stolen laptop and network administrators to respond by revoking the lost certificate.

### Security considerations: Group security

With the group authentication option, IKE PSK authentication is used with a KEY\_ID form of ID. Both the PSK and the KEY\_ID are derived from a group username and password. The derivation algorithm is kept secret by Avaya. In this form of authentication, there is also an Xauth authentication layer to prompt the user for a username and password and authenticate these credentials against a back end RADIUS server. During the Xauth phase, the client

provides the username and password in network packets which are to be protected by the IKE SA. An attacker who does not know the secret derivation algorithm cannot mount attacks on the exchange. An attacker who knew the secret derivation algorithm and the group username and password can mount man-in-the-middle attacks to observe usernames and passwords as legitimate users are logging in. Usually, the group username and password is not treated as a secret by administration and is rarely modified when the users of the system change (for example, through hirings or firings). Most of the security of this system depends upon the secrecy of the PSK/KEY\_ID derivation algorithm. However, this secret key derivation algorithm was published as [draft-mamros-pskeyext-00.txt](#). This renders the most popular authentication option the least secure.

The Avaya VPN client and Avaya VPN IRAS never rekey a phase1 SA. The original phase1 SA remains for the entire lifetime of the connection and only the phase2 SA is rekeyed. This deviates from the spirit of RFC standards which imagine that phase1 SAs are rekeyed with a frequency chosen by a site administrator for the purpose of defending against aging keys in cryptanalysis attacks.

If an attacker did find a very long lived Avaya VPN client to IRAS session and was able to cryptanalyze through the phase1 encryption keys, they would be able to observe the negotiations in Quick Mode channel. From this point, it must be assumed the attacker would also have the ability to start co-deriving the key material for the phase2 SAs themselves and gain the ability to observe data in the IPsec channel.

You can defend against this attack by enabling PFS with respect to keys. Enabling PFS has the effect of causing each new QuickMode negotiation to include the optional new Diffie-Hellman exchange. In the Secure Router configuration, this means enabling `pfs-group` in the `ipsec` subtree. Enabling `ipsec:pfs-group` becomes a strongly recommended best practice in this environment in spite of it being regarded as excessive in other normal environments that adhere to the spirit of the RFC standards. Therefore, `ipsec:pfs-group` is enabled by default in the Secure Router.

---

## Proprietary IKE Keepalive

On an Avaya VPN connection, the Avaya VPN IRAS does not send or respond to RFC 3706 Dead Peer Detection messages. A proprietary IKE keepalive is employed with behavior affected by some configuration variables.

Keepalives provide a twofold mechanism to:

- Define a certain time where the Avaya VPN IRAS can conclude that a connection has been lost, and reclaim connection resources.
- Allow an Avaya VPN client to keep a dialup connection going when VPN traffic is not flowing, when normally the ISP would hang up the call for non-activity.

On the Secure Router 2330/4134 side, keepalives can be enabled or disabled, and the interval and max retransmission can be configured. The Secure Router 2330/4134 sends keepalives based on these configuration parameters. The Secure Router 2330/4134 also sends these configuration parameters to the Avaya VPN client which, if using keepalive, uses these intervals.

The client has an option to enable or disable keepalives. The client sends keepalives only if both the Secure Router 2330/4134 and the client have it enabled. For interval and re-

transmissions, the client learns the configured interval and max-retransmissions from the Secure Router 2330/4134 and uses them.

---

## Standard Client Configuration Push

The Secure Router 2330/4134 IRAS can be configured to provide Avaya VPN clients with the following:

- an IP address from a configured address pool or from a RADIUS server
- a subnet mask
- primary and secondary DNS servers
- primary and secondary NBNS servers

When using RADIUS, the RADIUS server can also provide this information. The RADIUS server information is preferred, even if it is absent. If the RADIUS server is configured to allow the Secure Router 2330/4134 to provide configuration information, the RADIUS server signals so by sending the IP address 255.255.255.254 back for client configuration. The Secure Router 2330/4134 knows to provide an IP address to the client from its own local address pool.

Configure an Secure Router 2330/4134 to provide these parameters to clients under the `crypto contivity-iras ike policy client configuration tree`.

---

## Extended Client Configuration Push

In supporting simplicity of configuration of the Avaya VPN client, clients can have Avaya proprietary configuration pushed to them from the Avaya VPN IRAS at connection time. This is done in the Mode-CFG channel.

Avaya VPN Routers support more features than the Secure Router 2330/4134 Release 10.2. The Secure Router 2330/4134 will support additional features in future releases.

## Supported Extended Client Configuration Push features

The following table lists the configuration parameters that the Secure Router 2330/4134 can push out to VPN clients. To provide these parameters to clients, configure the appropriate parameters under the Secure Router 2330/4134 `crypto contivity-iras ike policy client configuration tree`.



**Table 12: Extended Client Configuration: Supported feature List**

Name	Description	Note	Secure Router 2330/4134 configuration parameters
Private-side-address	The IP address of any crypto trusted interface. Use a loopback interface as a recommended best practice. The Avaya VPN client drops the connection if this is not provided. And if the Avaya VPN client is being told to retrieve a banner-text it will socket connect to this address looking for one.	There is no such configuration requirement on the Avaya VPN Router line. Those routers are hard coded to know which interfaces were trusted and untrusted and they are able to assume a private-side-address without requiring the administrator to configure it. But on the Secure Router 2330/4134, each interface is configurable as to whether it is trusted or untrusted, thus the administrator must explicitly pick a private-side-address	private-side-address
Split Tunneling (bifurcation)	List of networks to route over VPN connection. Other networks should route out default ISP gateway without tunnel/security treatment.	Mutually exclusive with inverse-split-tunnel. If neither split-tunnel nor inverse-split-tunnel are configured then all traffic is compulsory tunneled over the VPN connection.	split-tunnel mode split-tunnel network
Inverse Split Tunnel	List of networks to exclude from routing over the VPN connection. Other network destinations are compulsory tunneled over the VPN.	Mutually exclusive with split-tunnel	split-tunnel mode split-tunnel network
Domain Name	List of DNS domains to search within.		domain-name

Name	Description	Note	Secure Router 2330/4134 configuration parameters
Failover Address / Failover List	List of up to three alternative Contivity IRAS targets for a client to use if it determines the primary is unreachable.	After contacting an Secure Router 2330/4134 once and receiving this config, a client remembers the list and uses the secondary targets if in the future it cannot contact the primary.	failover-list
No Keepalive	Grants permission to the client to disable keepalive. Keepalive processing is only disabled if the client acknowledges this parameter.	When multiple Avaya VPN clients are behind NAT, all clients can use the same user profile if both Keepalive and NAT Keepalive are enabled.	keepalive enable
Keepalive Interval	The number of seconds that MUST elapse before the client probes to check if the Contivity IRAS is still reachable.		keepalive interval
Keepalive Max Retransmissions	The maximum number of times the client should re-transmit a keepalive packet when detecting if the IRAS is still reachable. After the retransmissions are exhausted, the client SHOULD assume that the IRAS is no longer reachable.		keepalive max-retransmit
Password Stored on Client	If this attribute contains a non-zero value then the Avaya VPN client allows the user to store the VPN password in the local password cache. If	The default is disabled.	client-may-store-password

Name	Description	Note	Secure Router 2330/4134 configuration parameters
	this value is zero the client MUST NOT allow the user to save the password in any local password caching mechanism.		
Screen saver <ul style="list-style-type: none"> <li>• password required</li> <li>• activation time</li> </ul>	The maximum number of minutes that the screen saver on the client host can be set to before activation. If this value is greater than zero then the client MUST ensure that a password protected screen saver is enabled and set to activate in less than or equal to the specified number of minutes. The client MUST also ensure that the screen saver in use is password protected with a non-zero length password.	A default password is not required. The default time is 5 minutes.	client-screen-saver
Display Banner		Default = no	banner-enable
NAT Keepalive	Interval between NAT keepalives. The value zero means disabled.	Default = 0 (meaning off) When multiple Avaya VPN clients are behind NAT, all clients can use the same user profile if both Keepalive and NAT Keepalive are enabled.	nat-keepalive

## Non-supported Extended Client Configuration Push features

The following table lists the features that are not supported in Secure Router 2330/4134 Release 10.2. If you are considering replacing Avaya VPN Routers with Secure Routers, examine this list of non-supported features carefully.

**Table 13: Extended Client Configuration: Non-Supported feature List**

Name	Description	Limitations
Xauth passcode	This attribute is set by the client to communicate a Passcode associated with Token based authentication.	Secure Router 2330/4134 only supports RADIUS/ UserDB Xauth in Release 10.2.
Xauth message	This attribute is populated by the Contivity IRAS with text it receives from a token authentication server. The client displays the content of this attribute to the user.	Secure Router 2330/4134 only supports RADIUS/ UserDB Xauth in Release 10.2.
Xauth challenge	For use with SecureID auth forms	Secure Router 2330/4134 only supports RADIUS/ UserDB Xauth in Release 10.2.
Max Roaming Time	The maximum allowable time permitted to resume a connection interrupted by a loss of connectivity due to roaming.	No Mobility support on Secure Router 2330/4134.
Hands Free time	The Hands-Free time period indicates the duration to continually attempt to resume the VPN tunnel session for the time interval, persistent failover processing.	No Mobility support on Secure Router 2330/4134.
Contact Information		
Access Hours		
Call Admission Policy		
Forwarding Priority		
Number of Logins		
Password Management:		No Password management features on Secure Router 2330/4134.

Name	Description	Limitations
<ul style="list-style-type: none"> <li>• Max age</li> <li>• Min length</li> <li>• Alpha-numeric required</li> </ul>		
Static Address		
Idle Timeout		
Max failed login attempt to lock account		
Access Network Name		
Database Authentication (LDAP)		No LDAP server support on Secure Router 2330/4134.
client policy	List of policy rules: defines an allowed application or server that the client MAY use while the VPN is connected. The client MUST check this set of policy rules during VPN establishment and periodically while the VPN established. Any detected violation of these policies MUST cause the VPN to be disconnected. The client does enforcement.	
Quote of the Day server	IP address of a QOTD server which the client must connect to and retrieve a message from.	This is not implemented on a Contivity 1100.
IPX		
Session Number of Links		
RSVP		
User IP Address Source		
User Bandwidth Policy		
Tunnel Guard		
Forced Logoff time		
IPsec Idle timeout		
Client selection		

Name	Description	Limitations
<ul style="list-style-type: none"> <li>• Allowed Clients</li> <li>• Allow undefined Networks</li> </ul>		
Client Auto Connect		
Failover Tuning		
Client Dynamic DNS registration		
Client dynamic DNS domain		
Auto Dial Domain List	List of DNS domains used by the automatic connection functionality of the Avaya VPN client. On the Windows platform the Avaya VPN client stores this list in the registry and enables the auto-connect feature if it was not already enabled. Upon attempting to access any host with a suffix contained in the specified domain list, the auto-connect feature automatically launches the Avaya VPN client	
Auto Dial Network List	List of networks. On the Windows platform the Avaya VPN client stores this list in the registry and enables the auto-connect feature if it was not already enabled. Upon attempting to access any host address contained in the network list the auto-connect feature automatically launches the Avaya VPN client	
User address pool name		No need in Secure Router 2330/4134. However, Address pools are supported.
Filters		Not supported, in the Secure Router 2330/4134 however an administrator can use the firewall to write policies against the address pool.

## Extra IKE status codes

The following section describes Secure Router 2330/4134 support for additional IKE status codes.

### Invalid DH Group Message

The Avaya VPN client proposes IKE SA parameters with DH-Group-8 by default. DH-Group 8 is not supported on the Secure Router 2330/4134. However, the Avaya VPN client implements a mechanism for a server to suggest a DH-Group the server is willing to use. The Secure Router 2330/4134 makes use of this `invalid-dh-group` message to provide the suggestion to the client to use DH-Group-5. The Avaya VPN client then attempts to initiate a new proposal with DH-Group-5.

During a normal Avaya VPN client connection, the Secure Router 2330/4134 may log events about a failed connection with `no-proposal-chosen` in phase1 due to a mismatched DH-Group. However, in the logs, administrators can see immediate retries from the client with acceptable proposals.

To Avaya VPN client users, this is not noticeable.

### Non-supported feature list

The following table lists Avaya VPN features that are not supported in Secure Router 2330/4134 Release 10.2.

**Table 14: Non-supported feature list**

Feature	Description
Client Dynamic DNS	The ability to register clients with an internal DNS system as they connect.
Proxy LDAP authentication and authorization	The ability to relay LDAP authentication requests.
Clear connections based on IP or username	The ability to delete specific connections from the Secure Router CLI
Bandwidth management and QOS	The ability to rate limit or apply QOS policing per Avaya VPN connection
Access Controls	max failed attempts, time of day.
Link Certificates	Entrust specific PKI interoperability
Cross certification for certificates	(Canadian federal regulatory requirement)
4096 bit key size in certificates	Secure Router limit is 2048 in this release.

Feature	Description
Restrict client access based on source IP	
Initial contact	Secure Router respects Initial contact if sent from client, but does not send IC to the client.
Client Address redistribution	
LZS compression	
Client version control	The ability to require specific version of a client before allowing connection.
per group radius / LDAP	
Mobility support	
IKEv2 / MOBIKE	
EAP	
MS-CHAPv2	
Max number of Groups	
TunnelGuard	
Password Expiration / Change	
PPTP	
Firewall policies per user or per group.	
IKE Load balance message	

---

## Avaya VPN client Interoperability

This section describes behaviors of the Avaya VPN client 6.01 and Avaya VPN client 6.07 clients. To interoperate with these Avaya VPN clients, the Secure Router 2330/4134 IRAS supports these behaviors.

If an Avaya VPN client is proposing Aggressive Mode, it uses PSK authentication. If an Avaya VPN client is proposing Main Mode, it uses RSA-SIG authentication. Other combinations can be treated as errors.

All Avaya VPN clients always use IKE-ID as KEY\_ID. The data of the key ID is derived from the username. This alleviates the user from having to enter yet another variable into the Avaya VPN client GUI.



---

## Phase 1 proposals

The Avaya VPN client proposes these Phase 1 SA parameters:

Number	Encryption Alg	Key len	Hash alg	DH group
1.	AES	256	SHA	Group-8
2.	AES	128	SHA	Group-8
3.	AES	256	SHA	Group-5
4.	AES	128	SHA	Group-5
5.	3DES		SHA	Group-2
6.	3DES		MD5	Group-2
7.	<unknown>		SHA	Group-2
8.	DES		SHA	Group-1
9.	DES		MD5	Group-1
10.	<unknown>		SHA	Group-1

The Secure Router can be configured to interoperate with proposal numbers 3, 4, 5, 6, 8, and/or 9. Configure these Secure Router parameters under the `crypto contivity-iras ike policy <name> proposal <priority> tree`.

The Avaya VPN client v 6.01 does not send the lifetime attribute in the SA proposal for phase1. As an Avaya VPN IRAS, the Secure Router assumes the phase1 lifetime is unlimited and never rekeys the phase1.

---

## Phase 2 proposals

An Avaya VPN client sends these phase 2 proposals:

Number	Encryption Alg	Key len	Hash alg
1.	3DES		MD5
2.	3DES		SHA
3.	AES	128	MD5
4.	AES	128	SHA
5.	AES	256	MD5
6.	AES	256	SHA

The Secure Router can interoperate with any or all of these. Configure these Secure Router parameters under the `crypto contivity-iras ipsec policy <name> proposal <priority> tree`.

---

## Rekey Behavior

The Avaya VPN client rekey behavior is as follows:

- The Avaya VPN client never initiates a rekey operation.
- The Avaya VPN client also interprets any SA delete message (phase 1 or phase 2) as a signal to drop both phase1 and phase2 (drop all connection to the server).
- The Avaya VPN client also refuses to recognize a phase1 rekey because it is not associated with the current IPsec SA.

To interoperate with the Avaya VPN client behavior, the Secure Router operates as follows:

- The server is responsible for initiating rekey (even if the server is configured in a responder-only-mode for remote access).
- The server must never expire or allow its phase-1 SA to drop.
- The server must initiate only a phase2 rekey.

For information about the security properties related to this specialized form of rekeying, see [Security considerations related to authentication methods](#) on page 94.

---

## Mandatory client configuration parameters

There is a certain, minimum configuration under the `crypto contivity-iras ike policy <name> client configuration tree` which must be configured before a client can successfully connect. The required parameters are:

- `address-pool`: an Avaya VPN client requires an IP address from the server before maturing a connection.
- `private-side-address`: There is no such configuration requirement on the Avaya VPN Router line. Those routers are hard coded to know which interfaces are trusted and untrusted and they are able to assume a private-side-address without requiring the administrator to configure it. But on the Secure Router 2330/4134 each interface is configurable as to whether it is trusted or untrusted, thus the administrator must explicitly pick a private-side-address.

---

## Banner Text

The Avaya VPN client may attempt to acquire a text message to display to the user as a banner. The maturity of the tunnel is held pending the user's acceptance of the banner. The client

attempts to retrieve the banner text only if the banner-enable parameter has been set. The client queries for banner text at the IP address provided by the `private-side-address` parameter. So a `crypto trusted` interface configured with this address must be reachable through the VPN. Recommended best practice is to use a loopback interface to carry the private-side-address.

The SR provides banner text to the Avaya VPN client either from (1) the banner-text parameter or (2) the `/cf0/contivityBanner.txt` file on the flash system in that order of priority. If both are configured the banner-text is provided. If neither is configured, an empty string is provided as banner to the Avaya VPN client.

The banner-text parameter can hold 200 characters of data without new lines.

The `/cf0/contivityBanner.txt` file is limited in size to 5 kb.

Administrators can place a text file on the `/cf0/` flash card by any of the following methods:

- Use the file download command which starts an FTP client session to an FTP server you specify.
- Enable the FTP server with the `ftp` command and initiate a connection to the Secure Router 2330/4134 from any FTP client.
- Copy the file onto a compact flash card, insert that card into the Secure Router 2330/4134 `/cf1/` slot and copy the file from `/cf1/` to `/cf0/`.
- Copy the file onto a USB jump drive and insert that drive into the USB port on a Secure Router 2330/4134 and copy the file from `/usb/` to `/cf0/`.

---

## Avaya VPN client interactions with routing

Before Avaya VPN clients connect to the Secure Router 2330/4134 IRAS (as shown in [Figure 23: Remote user access to corporate network](#) on page 90), they have an IP address (perhaps behind a NAT) with access to the Internet. The clients receive this address from any local administrative entity. It can be assigned by an ISP, a DHCP server in a hotel or a coffee shop, or from a home network. This address is assigned to the physical Ethernet interface of the client.

During connection establishment, the Secure Router 2330/4134 provides a new IP address to the client from the administratively configured address pools. This new address is used by the client to create a new virtual interface. Now the client has two IP addresses and two interfaces, a physical and a virtual.

In the tunnel connecting the client to the SR-IRAS, the client physical address is the outer tunnel header and the virtual address is the inner tunnel header. The outer header traverses the Internet from the client to the Secure Router 2330/4134. The Secure Router 2330/4134 strips the outer header and decrypts the payload, exposing the inner packet with the virtual

address. This inner packet is forwarded by the Secure Router 2330/4134 to destinations in the corporate network.

When the packet reaches its destination (for example, a web server) the server sends a reply back to the client at the client virtual address. At this point, the corporate network inherits a responsibility to route the packet that is destined to the client virtual address out through the Secure Router 2330/4134. The corporate network must be informed that the route to reach the address pools on the Secure Router 2330/4134 must traverse through the Secure Router 2330/4134.

Network designers have several ways to accomplish this. Some examples include:

- Take note that the Secure Router itself creates connected, 32 bit routes for each client. If no other routers are involved in the corporate network, this is sufficient.
- Have the SR-IRAS also function as the IGP border router.
- Create a static route matching the address pools and redistribute the static route into an IGP like OSPF or RIP.

# Chapter 7: GRE and IPIP tunneling fundamentals

A tunnel is a logical interface that provides a framework for encapsulating passenger packets inside a transport protocol. GRE and IPIP are standards-based (RFC2784) (RFC1853) tunneling protocols that can encapsulate packets inside an IP tunnel, creating a virtual point-to-point link between routers at remote points over an IP network.

The advantage of using tunnels is that, while IPsec VPNs only function with IP unicast frames, GRE and IPIP are capable of handling the transportation of IP multicast traffic between two sites that only have IP unicast connectivity.

If encryption is required for a tunnel, you can enable IPsec transport mode over GRE/IPIP tunneling. This allows for the encryption and the transportation of multi-protocol traffic across the tunnel because both unicast and multicast IP packets appear to the IPsec protocol as IP unicast frame after GRE/IPIP tunneling.

The Avaya Secure Router 2330/4134 also supports a tunneling feature set for transitioning to IPv6, including IPv6 over manually-configured IPv4 tunnels, IPv6 over IPv4 GRE tunnels, and automatic 6to4 tunnels. These tunneling features provide a basic way for IPv6 hosts or islands to reach other IPv6 entities using IPv4 routing domains as the transport layer.

Multicast routing and unicast routing are supported on all tunnels, except automatic 6to4 tunnels.

---

## GRE and IPIP tunneling for IPv4

GRE and IPIP tunnels support the following features:

- tunnel protection: associates a tunnel interface with an IPsec profile. All traffic through the tunnel is encapsulated before it is encrypted.
- path MTU (PMTU) discovery: supported in order to avoid IP fragmentation. The DF bit from the inner IP header is copied to the outer IP header, allowing intermediate routers to fragment or not depending on the value of the DF bit. IP Fragmentation is supported for IP packets that exceed the MTU after insertion of the GRE/IPIP header.
- configurable TOS parameter: TOS bits from the inner (passenger) IP header are copied to the outer (transport) IP header. This allows the QOS DiffServ technology to operate on intermediate routers between GRE tunnel endpoints. Additionally the TOS bits are configurable
- routing protocols: can be enabled on the tunnel interface.
- multicast routing protocols: can be enabled on the tunnel interface.

In addition, GRE tunnels support the following additional features:

- tunnel keepalive: sends keepalive packets to keep track of the tunnel end points and take down the line protocol of the GRE tunnel interface if the far end becomes unreachable
- optional data sequencing: dropping of out of order data grams

---

## IPIP

IP in IP encapsulation differs from GRE in that it does not insert its own special glue header between IP headers. Instead, the original IP Header is retained, and simply wrapped in another standard IP header

---

## GRE

Generic Routing Encapsulation (GRE) is a standards-based (RFC1701, RFC2784) tunneling protocol that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link between routers at remote points over an IP network. A tunnel is a logical interface that provides a way to encapsulate passenger packets inside a transport protocol. By connecting multiprotocol subnetworks in a single-protocol backbone environment, IP tunneling using GRE allows network expansion across a single-protocol backbone environment. GRE tunnels can be used for unencrypted traffic.

---

## Tunnel protection

IPsec transport mode is used to provide protection for packets that already encapsulated (or tunneled) using other protocols. Both IPIP and GRE can operate with IPsec to provide tunnel protection. IPsec can be used in transport mode with these protocols and it can provide security for these packets using ESP and/or AH.

The Secure Router 2330/4134 only supports tunnel protection for IPv4 tunnels.

---

## VLAN over GRE

VLAN traffic can be carried over Ipv4 Generic Route Encapsulation (GRE) tunnels using the standard GRE tunnels. The tunnels carry VLAN as the passenger protocol with GRE as the carrier protocol and Ipv4 or Ipv6 as the transport protocol. GRE has a protocol field that identifies the passenger protocol.

The entry tunnel end point (the encapsulating node) encapsulates the VLAN packet with a GRE-IPv4 header using the configured v4 tunnel source and tunnel destination IP addresses. The exit tunnel end point removes the outer GRE-IPv4 header and processes the received VLAN packet.

Tunnel protection is not available for a GRE tunnel carrying VLAN traffic.

---

## PCAP over GRE

Beginning with Release 10.2, Secure Router 2330/4134 supports packet capture (PCAP) over GRE tunnels.

---

## IPv6 over IPv4 tunneling

---

### IPv6 over manually-configured IPv4 tunnels

A manually configured IPv6 over IPv4 tunnel is equivalent to a permanent link between two IPv6 domains over an IPv4 backbone. An IPv6 address is manually configured on a tunnel interface, and manually configured IPv4 addresses are assigned to the tunnel source and the tunnel destination. The host or router at each end of a configured tunnel must support both the IPv4 and IPv6 protocol stacks. Manually configured tunnels can be configured between border routers or between a border router and a host.

The entry tunnel end point (the encapsulating node) encapsulates the IPv6 packet with an IPv4 header with the configured IPv4 tunnel source and tunnel destination IP addresses. The exit tunnel end point (the de-capsulating node) removes the outer IPv4 header and forward or consume the received IPv6 packet.

---

### IPv6 over IPv4 GRE tunnels

IPv6 traffic can be carried over IPv4 Generic Route Encapsulation (GRE) tunnels using the standard GRE tunneling technique. As in IPv6 manually-configured tunnels, GRE tunnels are links between two points, with a separate tunnel for each link. The tunnels carry IPv6 as the passenger protocol with GRE as the carrier protocol and IPv4 as the transport protocol. GRE has a protocol field that identifies the passenger protocol.

The entry tunnel end point (the encapsulating node) encapsulates the IPv6 packet with a GRE-IPv4 header using the configured IPv4 tunnel source and tunnel destination IP addresses. The exit tunnel end point removes the outer GRE-IPv4 header and processes or forwards the received IPv6 packet.

---

## Auto 6to4 tunneling

Auto 6to4 tunneling is a dynamic way to deploy tunnels between sites made up of IPv6 nodes. Tunneling of IPv6 packets is done dynamically, using the destination IPv6 address of a packet originating from the IPv6 node. Auto 6to4 encapsulates the IPv6 packet in IPv4 and uses the IPv4 routing domain. The destination IP address of the tunnel does not need to be manually preset. The IPv4 address embedded in the 2002:: /16 prefixed destination IPv6 address is used to find the other end of the automatic tunnel.

The key difference between automatic 6to4 tunnels and manually-configured tunnels is that the tunnel is not point-to-point; rather, it is point-to-multipoint.

6to4 prefixes use the 2002:: /16 address space assigned by IANA. A globally unicast IPv4 assigned to the 6to4 router is converted to hexadecimal and appended to the 2002:: /16 prefix.

Routing protocols are not supported on auto 6to4 tunnels.

---

## Standards compliance

The Secure Router 2330/4134 implementation of GRE and IPIP tunneling complies with the following RFCs:

- RFC 1853, IP in IP Tunneling
- RFC 2784, Generic Routing Encapsulation (GRE)
- RFC 2890, Key and Sequence Number Extensions to GRE
- RFC 2893, Transition Mechanisms for IPv6 Hosts and Routers
- RFC 3056, Connection of IPv6 Domains via IPv4 Clouds



# Chapter 8: PPPoE client fundamentals

PPPoE (RFC 2516) is a commonly used application in the deployment of DSL. One of the main advantages of PPPoE is that it offers authentication based on Password Authentication Protocol (PAP) or Challenge Handshake Authentication Protocol (CHAP).

With the Avaya Secure Router 2330/4134 the PPPoE client must be deployed with a DSL modem providing a connection to a PPPoE server for access to Internet resources.

The main purpose of PPPoE is to serve as a backup and fail-over solution. When the primary connectivity goes down, traffic can switch over to a backup interface, in this case the virtual PPPoE interface. In this scenario, a PPPoE client session is established with a PPPoE server and traffic is routed through this path until the primary connectivity is restored.

The PPPoE client only transmits traffic if it is up and running and a route lookup picks a route pointing to the PPPoE interface.

For PPP to run over Ethernet, it needs to learn the Ethernet address of the remote peer, and establish a session ID. In order to do this the PPPoE protocol defines a discovery stage, followed by a session stage. In the discovery stage the host acts as a client and the remote access concentrator acts as a server. This allows the host to learn the Ethernet address of the remote peer and to set up a session ID. After the discovery stage, the session stage begins with the normal PPP LCP and IPCP negotiation.

On the Secure Router 2330/4134 the PPPoE client supports the following features

- The client connection to the PPPoE server is established when the client is configured.
- Traffic is routed to the PPPoE based on route selection. When the optimal route becomes inactive and a route pointing to the PPPoE interface is active, traffic switches over to the PPPoE interface. Traffic stops using the PPPoE interface when a more optimal route becomes active.
- The PPPoE client can learn its IP address from the remote PPPoE server.
- The PPP session sends keepalives at a regular interval to detect PPPoE server failure.
- IPsec over PPPoE is supported.

The following considerations are important in configuring PPPoE:

- This implementation only supports client PPPoE.
- PPPoE drops any data packets it receives if it is not the primary interface.
- There is no ACL or VLAN support on PPPoE interfaces.
- PPPoE cannot run on an Ethernet interface configured for VLAN encapsulation.
- QoS is not supported on PPPoE virtual access interface in this release, but QoS can be configured on the parent Ethernet port.
- The PPPoE client implementation cannot be used to connect to two DSL connections for load sharing.
- Neither unicast nor multicast routing protocols are supported on PPPoE.

- When the IP address is negotiated with the PPPoE server, the PPPoE client assumes that the subnet mask for the IP address is 32 bits wide.
- The PPPoE virtual access interface is automatically configured in the untrusted internet security zone. Configure the Ethernet interface on which PPPoE is running in the internet security zone. To allow traffic to pass through the firewall, at least one other router interface must be configured in a trusted firewall zone (for example, corp).

---

## Standards compliance

The Secure Router 2330/4134 implementation of PPPoE supports the PPPoE client implementation specified in RFC 2516 - A Method for Transmitting PPP over Ethernet (PPPoE).

# Chapter 9: Authentication, Authorization, and Accounting fundamentals

AAA provides a modular way of performing the following services:

- Authentication
- Authorization
- Accounting

---

## Authentication

Authentication determines how a user is identified prior to being allowed access to the network and network services. You configure AAA authentication by defining a named list of authentication methods, and then applying that list to various interfaces. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific interface before any of the defined authentication methods can be performed. The only exception is the default method list. The default method list is automatically applied to all interfaces if no other method list is defined. A defined method list overrides the default method list.

All authentication methods, except for local, line password, and enable authentication, must be defined through AAA.

Avaya Secure Router 2330/4134 supports the following authentication methods:

---

## PAP authentication

Password Authentication Protocol (PAP) is a simple authentication protocol used to authenticate a user to a remote access server or Internet service provider (ISP). Almost all network operating system remote servers support PAP. PAP transmits unencrypted ASCII passwords over the network and is therefore considered insecure.)

---

## CHAP authentication

Challenge Handshake Authentication Protocol (CHAP) is an authentication scheme used by Point to Point Protocol (PPP), Telnet, and SSH to validate the identity of remote clients. CHAP periodically verifies the identity of the client by using a three-way handshake.

---

## RADIUS

Remote Access Dial-In User Services (RADIUS) is a distributed client/server system that assists in securing networks against unauthorized access, allowing a number of communication servers and clients to authenticate user identities through a central database. The database within the RADIUS server stores information about clients, users, passwords, and access privileges, protected with a shared secret. RADIUS authentication is a fully open and standard protocol defined by RFC 2865.

A RADIUS application has two components:

**RADIUS server**—a computer equipped with server software (for example, a UNIX workstation) that is located at a central office or campus. It has authentication and access information in a form that is compatible with the client. Typically, the database in the RADIUS server stores client information, user information, password, and access privileges, protected with a shared secret.

**RADIUS client**—a switch, router, or a remote access server equipped with client software, that typically resides on the same LAN segment as the server. The client is the network access point between the remote users and the server.

RADIUS authentication allows a remote server to authenticate users attempting to log on to the router from the local console or Telnet.

---

## TACACS

Terminal access controller access control system (TACACS+) is a security application implemented as a client/server-based protocol that provides centralized validation of users attempting to gain access to a router or network access server. TACACS+ differs from RADIUS in two important ways:

- TACACS+ is a TCP-based protocol.
- TACACS+ uses full packet encryption, rather than just encrypting the password (RADIUS authentication request).

TACACS+ encrypts the entire body of the packet but uses a standard TACACS+ header. TACACS+ separates authentication, authorization, and accounting services. This means that you can selectively implement one or more TACACS+ services.

### **Secure Router 4134 TACACS+ Support for Cisco Privilege Levels**

The ability of support for Cisco Privilege levels simplifies the setup of AAA authorization of commands when using TACACS+ Server. Cisco defined 16 possible privilege levels in which 3 are set up by default. This release supports all the default levels that Cisco supports which are level zero, one and fifteen. For all other TACACS+ Privilege levels the user will have the same user level as the default user. The TACACS+ privilege level maps to a corresponding user levels on the SR 4134 which each user can see by issuing the CLI command **show**

**whoami**. The corresponding mapping between the TACACS+ Privilege levels and SR 4134 user levels are as follows:

TACACS+ Privilege Level	SR 4134 User Level
15	1 (super user)
1	4 (default)
0	3

The following table shows the CLI commands each user level supports:

User Level	Privilege Name	Definition
1 (highest)	PRIVILEGE_ADMIN	Admin Level Can access any command and configure any feature in the router, including user configuration and administration
2	PRIVILEGE_CONFIGURE	Configure Level Can access any command and configure any feature in the router, except user configuration and administration
3	PRIVILEGE_TEST	Test Level Every command that level 4 can execute plus clear arp, show conf, show run, show start, show user, show file.
4 (default)	PRIVILEGE_NORMAL	Normal Level Can only enter ping, trace, mtrace telnet, and show commands except for: (show conf, show run, show start, show user, show file, show ftp)

### Setting up TACACS+ Server for Privilege Levels

Simple setup for the TACACS+ server is to set up a separate group for each privilege level on your TACACS+ server and then assign each user to the appropriate group. The following figure shows a portion of the tac\_plus.cfg file used by a Freeware version of TACACS+ server on Linux.

```
group = configure {
service = exec {
priv-lvl = 15
}
}
group = test {
```

```

service = exec {
  priv-lvl = 0
}
group = group1 {
  service = exec {
    priv-lvl = 1
  }
}
user = admin {
  login = file /etc/passwd
  member = configure
}
user = tester {
  login = file /etc/passwd
  member = test
}
user = user1 {
  login = file /etc/passwd
  member = group1
}

```

On the SR 4134 under the AAA configuration section you need to specify that the AAA authentication and authorization will use TACACS. The following commands need to be specified under the AAA section with the appropriate protocols and other services.

```

SR4134/configure/aaa#authentication login default tacacsSR4134/
configure/aaa#authentication protocols default asciiSR4134/
configure/aaa#authorization commands default tacacs

```

---

## EAP IEEE 802.1X

The Extensible Authentication Protocol over LAN (EAPoL) or IEEE 802.1X is a port-based network access control protocol that allows you to set up network access control on internal LANs.

EAPoL provides security in that it prevents users from accessing network resources before they are authenticated. Without this authentication, users could access a network to assume a valid identity and access confidential material or launch denial-of-service attacks.

EAP allows the exchange of authentication information between any end station or server connected to the switch and an authentication server, such as a RADIUS server. The EAPoL-based security feature operates in conjunction with a RADIUS-based server to extend the benefits of remote authentication to internal LAN clients.

You can use EAPoL to set up network access control on internal LANs and to exchange authentication information between any end station or server connected to the Secure Router and an authentication server (such as a RADIUS server). This security feature extends the benefits of remote authentication to internal LAN clients. For example, if a new client PC fails the authentication process, EAPoL prevents it from accessing the network.

---

## RSA SecureID

This section describes the Secure Router support for two factor authentication using RSA SecureID.

Previously, the Secure Router authenticated telnet and console users through a username and a password only (single factor authentication). The privilege level associated with each username and password could be configured on the router itself or on a AAA server (RADIUS or TACACS+).

Single factor authentication relies on the user to take the necessary precautions to maintain security, for example, by creating a strong password and ensuring that no one else can access it. For applications that require greater security, it can be advisable to implement more complex systems, such as two-factor authentication.

With two-factor authentication, the user must specify a username, security PIN, and an authentication code from a physical token code generator. The generated token code provides an additional means of identification for the user.

A token code generator is a portable piece of hardware that generates an authentication code based on a factory-encoded key (or seed) at fixed intervals using a built-in clock. Each seed is unique, and is also loaded onto the corresponding authentication server. The server uses the seed to compute the number the token code generator is showing at any given moment. When the user enters the displayed token code, the server checks the user-entered value against its own computed value, and allows or denies access accordingly.

To achieve two factor authentication, the Secure Router can support authentication of remote users through an ACE server.

With an ACE server, the parameters for the user authentication are as follows:

- Username
- 4 – 8 digit security PIN
- Token code from the RSA SecureID token code generator equipment

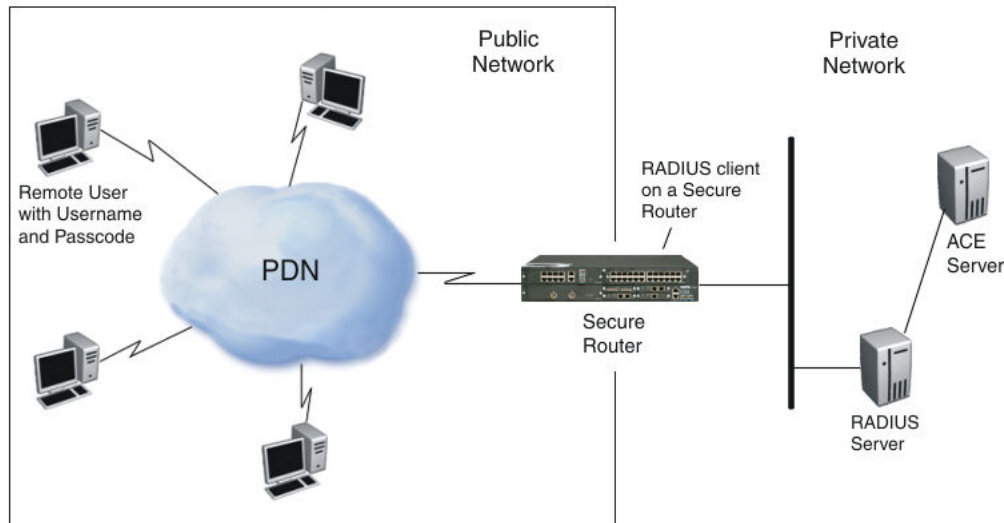
The token code is generated from the RSA SecureID token code generator equipment provided by the RSA organization. The username and the serial number of the token code generator equipment are tightly bound at the ACE server.

To authenticate with the ACE server using the Secure Router, the remote user must enter a username and passcode to be authenticated by the ACE server. The passcode is a combination of a security PIN and the token code from the token code generator.

### Feature Description

The Secure Router support of the ACE server requires a RADIUS server as an intermediate communicator between the Secure Router and the ACE server.

The following diagram shows the Secure Router deployment scenario.



After establishing a telnet connection to the Secure Router, the remote user must enter a username and passcode to be authenticated by the ACE server.

**Important:**

The passcode is a combination of the security PIN plus token code from the token code generator equipment provided by the RSA SecureID organization.

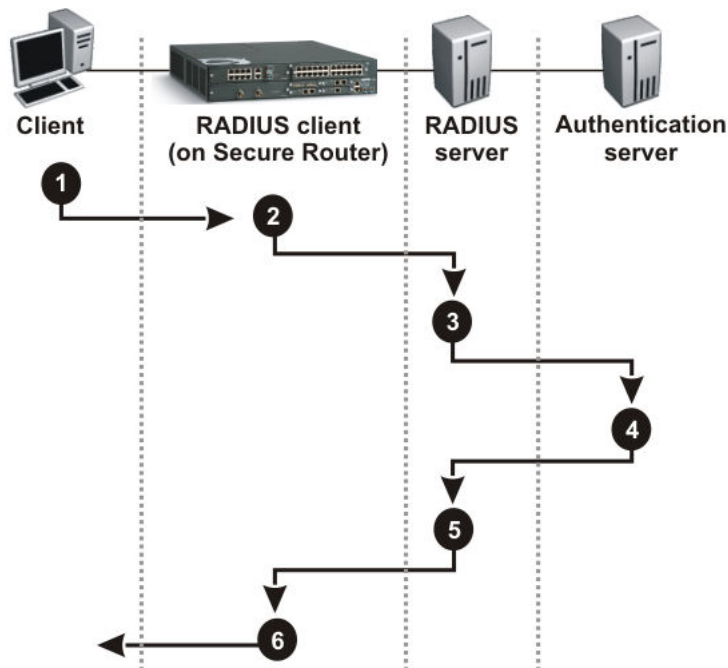
When the Secure Router receives the username and passcode entered by the user, the RADIUS client on the Secure Router generates an access request packet to send to the RADIUS server.

The RADIUS server is configured to redirect the request to the ACE server. If the username and passcode entered by the user are valid, the ACE server sends its acceptance packet to the RADIUS server. In this case, the RADIUS server directs an access accept packet to the RADIUS client.

If the username and the passcode entered by the user are not valid, then the RADIUS server sends an access reject packet to the RADIUS client.

The following diagram shows the round trip communication between the end user and the ACE server.





### RSA SecureID authentication modes

There are two modes with RSA SecureID authentication.

1. New pin mode
2. Next token mode

#### New pin mode:

When a new user is configured on the ACE server, the administrator can keep the user under new pin mode.

This gives the user the option to change the PIN by providing a default PIN.

When the user establishes a telnet connection to the router and enters the username and passcode (default PIN given by the administrator plus token code), a challenge message is prompted giving an option to change the PIN.

#### Important:

If the ACE server is in new pin mode, and no PIN is configured for the user, then the input from the user is the username and the token code only. A challenge message is then prompted providing an option to create a new pin.

#### Next token mode:

If a user establishes a telnet connection to the router and enters a valid PIN but the wrong token code, after three such consecutive attempts, the user is moved to next token mode.

Under next token mode, if the user enters a valid username and passcode, a challenge message is prompted to the user for the next valid token code.

The Secure router supports both new pin mode and next token mode.

### Configuration steps

To support the ACE Server, there are no new configuration commands on the Secure Router. The following steps describe the configuration required on the Secure Router to support the ACE server:

1. Configure the RADIUS server parameters on the Secure Router.
2. Configure the AAA login method and protocol (login method radius with pap protocol)
3. Enable AAA.
4. Apply the method and protocol to the interface.

For information on configuring the RADIUS server and ACE server, see the appropriate documentation for these products.

### Limitations

Currently, authentication through ACE server - RSA SecureID is supported for telnet and console logins only. RSA SecureID does not currently support SSH or IPSec authentication.

---

## Standards compliance

The Secure Router 2330/4134 supports the following standards:

#### RADIUS

- RADIUS Basic: RADIUS RFC2058, RFC2138 (obsoleted by 2865) , RFC2865 (obsoletes 2138))
- RADIUS Extension: RADIUS Accounting (RFC2139, RFC2866)

The following MIBs are also supported:

- RADIUS Authentication Client MIB (RFC2618)
- RADIUS Accounting Client MIB (RFC2620)

#### TACACS+

- TACACS+ Protocol Version 1.76

---

## Authorization

Authorization provides the method for remote access control, including one-time authorization or authorization for each service, each user account list and profile, user support, and support of SSH and Telnet.

Authorization works by assembling a set of attributes that describe what the user is authorized to perform. These attributes are compared to the information contained in a database for a

given user and the result is returned to AAA to determine the user's actual capabilities and restrictions. The database can be located locally on the router or it can be hosted remotely on a RADIUS or TACACS+ security server. Remote security servers, such as RADIUS and TACACS+, authorize users for specific rights by associating attribute-value (AV) pairs, which define those rights with the appropriate user. All authorization methods must be defined through AAA.

As with authentication, you configure AAA authorization by defining a named list of authorization methods, and then applying that list to various interfaces.

---

## Accounting

Accounting provides the method for collecting and sending security server information used for billing, auditing, and reporting, such as user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

Accounting enables you to track the services users are accessing as well as the amount of network resources they are consuming. When AAA accounting is activated, the network access server reports user activity to the RADIUS or TACACS+ security server (depending on which security method you have implemented) in the form of accounting records. Each accounting record is comprised of accounting AV pairs and is stored on the access control server. This data can then be analyzed for network management, client billing, and auditing. All accounting methods must be defined through AAA. As with authentication and authorization, you configure AAA accounting by defining a named list of accounting methods, and then applying that list to various interfaces

---

## TACACS accounting

The Secure Router 2330/4134 supports Terminal Access Controller Access Control System (TACACS) accounting. This feature allows an administrator to audit user activity on a router at any date or time or both. TACACS accounting details what commands were issued by a particular user.

The TACACS accounting system tracks and stores Attribute Value data on a TACACS accounting server. This accounting data includes details such as user name, the user's IP address, a timestamp and the activity—perhaps a Login or execution of a particular command. The data can then be analyzed for user activity on a router at any date or time. For example, when a user connects to an interface remotely through Telnet or SSH using the correct username and password, a log is written and can be viewed on the TACACS server.

All accounting methods must be defined through Authentication Authorization Accounting (AAA). Much like AAA, TACACS accounting is configured through the definition of a named list of accounting commands with specific methods, and then applying this list to one or more interfaces.

The two main TACACS accounting commands are as follows:

- **network**—If applied to an interface, enables accounting for user login and logout.
- **commands**—If applied to an interface, enables accounting for all commands executed by a user.

The methods of TACACS accounting are as follow:

- **stop-only**—If specified, sends a notice to stop record accounting at the end of the specified activity.
- **start-stop**—If specified, sends a notice to start record accounting after a process begins and sends a notice to stop record accounting at the end of the specified activity. This allows the requested user process to begin even if the start accounting record was not acknowledged by the accounting server.
- **wait-start**—If specified, sends a notice to start and stop accounting to the accounting server. In this scenario, the user service does not begin until the start accounting record is acknowledged.

**Note:**

If you create an accounting method list with a list name of "default", all interfaces uses this list without applying it on an interface. You can override this "default" list only when you create an explicit method list and apply it to the interface.

# Chapter 10: SSH2 fundamentals

This chapter contains information to help you understand SSH version 2.0 (SSH2).

---

## SSH2 features

The Secure Shell (SSH) server is designed as a secure alternative to well known applications like Telnet, rlogin, RCP, RSH and FTP. The Avaya Secure Router 2330/4134 supports an SSH server to provide authentication, confidentiality and integrity. It is a protocol for secure remote login and other secure network services over an insecure network. It also supports compression.

The SSH version 2.0 server has a layered architecture.

The Transport Layer Protocol provides server authentication, confidentiality, and integrity. It can also optionally provide compression. The authentication in this protocol level is host based.

The User Authentication Protocol runs over the transport layer protocol and authenticates the client-side user to the server.

The Connection Protocol runs over the user authentication protocol and multiplexes the encrypted tunnel into several logical channels.

Secure shell implementation typically consists of SSH server, SSH client, SCP client and SFTP client.

The SSH server listens for connections (on port number 22) from client machines. Whenever it receives connection, it performs authentication and starts serving the client.

The SSH client is used to log into another machine or to execute commands on the other machine.

The secure copy (SCP) client is used to securely copy files from one machine to another.

The secure FTP (SFTP) client is used for secure interactive file transfer, similar to FTP.

The Secure Router 2330/4134 supports the SSH server, but not SSH clients.

---

## SSH ciphers

Ciphers are methods for encrypting and decrypting data. There are two classes of key-based encryption algorithms, symmetric (or secret-key) and asymmetric (or public-key) algorithms. While symmetric algorithms use the same key for encryption and decryption, the asymmetric algorithms use different keys for encryption and decryption (a private and public key pair). Here the decryption key cannot be derived from the encryption key. The list of symmetric algorithms supported for Secure Router 2330/4134 are as follows:

**Table 15: SSH ciphers**

Name	Description
3des-cbc	3 key DES in CBC mode
blowfish-cbc	Blowfish in CBC mode
aes128-cbc	AES, CBC mode, 128-bit key
aes192-cbc	AES, CBC mode, 192-bit key
aes256-cbc	AES, CBC mode, 256-bit key

---

## SSH MAC algorithms

The Message Authentication Code algorithms are usually hash functions which compress the bits of a message to a fixed-size hash value in a way that distributes the possible messages evenly among the possible hash values. A cryptographic hash function does this in a way that makes it extremely difficult to come up with a message that would hash to a particular hash value. The MAC algorithms supported for Secure Router 2330/4134 are as follows:

**Table 16: SSH MAC algorithms**

Name	Description
hmac-sha1	HMAC-SHA1 (digest length = key length = 20)
hmac-sha1-96	first 96 bits of HMAC-SHA1 (digest length = 12, key length = 20)
hmac-md5	HMAC-MD5 (digest length = key length = 16)
hmac-md5-96	first 96 bits of HMAC-MD5 (digest length = 12, key length = 16)

---

## SSH compression

SSH uses GNU ZLIB (LZ77) for compression. The ZLIB compression is described in RFC 1950 and in RFC 1951. By default, compression is enabled.

---

## SSH key exchange methods

For the symmetric ciphers to work, shared secrets have to be created and communicated over insecure communications channel. The key exchange algorithms are based on modulo arithmetic and are used to exchange keys between two systems that do not share any mutual keys, securely. The key exchange algorithms supported for Secure Router 2330/4134 are as follows:

**Table 17: SSH key exchange methods**

Name	Description
diffie-hellman-group1-sha1	DH
diffie-hellman-groupexchange-sha1	DH Group exchange draft draft-ietf-secsh-dh-group-exchange-03.txt

---

## SSH public key algorithms

Public key algorithms are asymmetric key algorithms, which rely on two keys: namely the public key and the private key. Data encrypted with one key can be decrypted with the other. It is nearly impossible to derive the private key from the public key. The public key algorithms supported for Secure Router 2330/4134 are as follows:

**Table 18: SSH public key algorithms**

Name	Description
ssh-dss	Simple DSS for signature
ssh-rsa	Simple RSA for signature

---

## SSH user authentication methods

The SSH client is authenticated to the SSH server using one of the following methods:

**Table 19: SSH user authentication methods**

Name	Description
public key	Public key authentication
password	Password based authentication

---

## SSH public key file formats

The SSHv2 protocol drafts specify a standard format for storing the public keys. In the Secure Router 2330/4134 keys are generated in the openssh format. This can be converted to the secsh format with the convert command:

**Table 20: SSH public key file formats**

Name	Description
openssh	openssh format The key generation command will generate public key file in this format
secsh	draft-ietf-secsh-publickeyfile-03.txt Use key conversion command to convert between openssh and secsh formats

---

## Standards compliance

The Secure Router 2330/4134 implementation of SSH2 supports the following standards:

- draft-ietf-secsh-architecture-13.txt: SSH Protocol Architecture
- draft-ietf-secsh-transport-15.txt: SSH Transport Layer Protocol
- draft-ietf-secsh-userauth-16.txt: SSH Authentication Protocol
- draft-ietf-secsh-connect-16.txt: SSH Connection Protocol
- draft-ietf-secsh-filexfer-04.txt: SSH File Transfer Protocol
- draft-ietf-secsh-dh-group-exchange-03.txt: Diffie-Hellman Group Exchange



- draft-ietf-secsh-fingerprint-01.txt: SSH Fingerprint Format
- draft-ietf-secsh-publickeyfile-03.txt: SSH Public Key File Format
- draft-ietf-secsh-assignednumbers-01.txt: SSH Protocol Assigned Numbers



# Chapter 11: Firewall and NAT configuration

This chapter describes how to perform Firewall and NAT configurations.

---

## Configuring global properties

Refer to the following sections to configure global firewall properties.

---

## Configuring global ALGs

Choose the ALGs to enable on the router.

All firewall ALGs are disabled by default.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`
3. To specify global ALG configuration, enter:  
`algs`
4. To enable or disable the specified ALGs, enter:  
`[no] {<alg>|enable-all|enable-typical}`

**Table 21: Variable definitions**

Variable	Value
<alg>	aim: enable/disable aim aimudp: enable/disable nntp cuseeme: enable/disable cuseeme dns: enable/disable dns ftp: enable/disable ftp gatekeeper: enable/disable gatekeeper h323: enable/disable h323 icq: enable/disable icq ike: enable/disable nntp ils: enable/disable ils ils2: enable/disable nntp irc: enable/disable irc l2tp: enable/ disable l2tp msgtcp: enable/disable msgtcp msgudp: enable/ disable msgudp msn: enable/disable msn mszone: enable/ disable mszone n2p: enable/disable n2p n2pe: enable/

Variable	Value
	disable n2pe netbios: enable/disable nntp nntp: enable/disable nntp pcanywhere: enable/disable pcanywhere pptp: enable/disable pptp rpc: enable/disable rpc rtsp554: enable/disable rtsp rtsp7070: enable/disable rtsp7070 sip [port <1-65535>]: enable/disable SIP ALG on UDP port 5060. (Optionally, specify an alternate UDP port to use for SIP ALG.) sip-tcp: enable/disable SIP ALG on TCP port 5060. smtp: enable/disable smtp sql: enable/disable sql tftp: enable/disable tftp web: enable/disable web
enable-all	Enables all ALGs.
enable-typical	<p>This option enables only a specific set of ALGs as follows</p> <ul style="list-style-type: none"> <li>• aim</li> <li>• aimudp</li> <li>• ftp</li> <li>• ike</li> <li>• msn</li> <li>• pptp</li> <li>• rpc</li> <li>• rtsp554</li> <li>• rtsp7070</li> <li>• smtp</li> <li>• tftp</li> <li>• web</li> </ul> <p>The remaining ALGs (tftp, gatekeeper, msnudp, dns, n2p, pcanywhere, sql, msgtcp, irc, n2pe, ils, cuseeme, mszone, ils2, nntp) are in the disabled state.</p>
[no]	Disables the specified ALGs.

## Configuring the DNS ALG

The DNS ALG is used when the DNS client in the untrusted side wants to access the DNS server behind NAT in the trusted side.

A DNS client in the untrusted side sends a "DNS Standard Query" to the Secure Router. The Secure Router receives the DNS query with the destination port 53. The Secure Router translates the IP header based on the reverse NAT policy. When the response comes from the DNS server (that is present in the trusted side), the Secure Router translates the header

based on the reverse NAT policy, and the DNS payload is translated from the private IP record to the global IP record, which is taken from the DNS pool database.

A DNS client in the untrusted side sends a "DNS Reverse Query" to the Secure Router. The Secure Router translates the IP header based on the reverse NAT policy, and the DNS payload is translated from the global IP record to the private IP record, which were added using the CLI. When the response comes from the DNS server (that is present in the trusted side), the Secure Router translates the header based on the reverse NAT policy, and the DNS payload is translated from the private IP record to the global IP record, which is taken from the DNS pool database.

To translate the IP address in the DNS payload, you must follow the configuration in the next section.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`
3. To specify global ALG configuration, enter:  
`algs`
4. To specify the DNS ALG configuration, enter:  
`dns`
5. To enable the DNS ALG, enter:  
`enable`
6. To configure the DNS pool, enter:  
`pool <pool-name> <private-ip> <global-ip>`
7. To display the information of the static pool name specified, enter:  
`show firewall dns-alg translate-pool <pool-name>`
8. To display the information of all static pool names configured, enter:  
`show firewall dns-alg translate-pool`

---

## Configuring global bypass trusted

Configure the firewall to bypass processing of traffic from trusted to trusted interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure global bypass trusted, enter:

```
[no] bypass-trusted
```

**Table 22: Variable definitions**

Variable	Value
[no]	Disables bypass trusted.

## Configuring global DOS protection

Configure global Denial of Service (DOS) protection to protect the network against intrusion attempts such as SYN attacks, Win-nyke attacks, and IP sequence number spoofing.

By default, all DOS protection checks are disabled except for SYN flooding, ICMP error, and DNS replay.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify global DOS protection configuration, enter:

```
dos-protect
```

4. To enable or disable desired DOS protection options, enter:

```
[no] <dos-protect-option>
```

**Table 23: Variable definitions**

Variable	Value
<dos-protect-option>	<i>enable-all</i> —Enables/disables all DOS protect checks.
	<i>dns-replay-attack</i> —Enables/disables DNS replay attack check. A DNS replay attack occurs when an individual intercepts traffic, analyzes the captured packets and obtains authentication information. The individual can then use this information to gain access to other systems by reinserting the authenticated packets on the Internet and replaying them.

Variable	Value
	<p>When this command is enabled, the DNS connection limit is 2000.</p> <p>By default, this option is enabled.</p>
	<p><i>ftp-bounce</i>—Enables/disables FTP bounce check.</p> <p>In a bounce attack, the hacker uploads a file to the FTP (File Transfer Protocol) server and then requests this file to be sent to an internal server. The file can contain malicious software that destroys data, or it can contain a simple script that executes instructions on the internal server that uses up all the memory and CPU resources.</p> <p>By default, this option is disabled.</p>
	<p><i>icmp-error</i>—Enables/disables ICMP error check.</p> <p>The <i>icmp-error</i> attacks target ICMP (Internet Control Message Protocol) error reporting system. By constructing packets that generate ICMP error responses, an attacker can overwhelm a server's incoming network and cause the server to overwhelm its outgoing network with ICMP responses.</p> <p>By default, this option is enabled.</p>
	<p><i>ip-unaligned-timestamp</i>—Enables/disables IP unaligned timestamp.</p> <p>Provides support for an unaligned IP timestamp check. Some operating systems crash if they receive a frame with the IP timestamp option not aligned on a 32-bit boundary.</p> <p>By default, this option is disabled.</p>
	<p><i>mime-flood</i>—Enables/disables MIME flood check.</p> <p>The MIME (Multipurpose Internet Mail Extensions) flood attack is possible on a web server. Here the attacker keeps sending numerous request headers of extremely long lengths to the target web server. Over time (and with enough headers), remote attackers can crash the web server or consume massive CPU resources, memory, and so on.</p> <p>By default, this option is disabled.</p>
	<p><i>source-routing</i>—Enables/disables source routing check.</p> <p>After enabling source routing check, the firewall filters out all the datagrams with the strict or loose source routing option enabled.</p> <p>By default, this option is disabled.</p>
	<p><i>syn-flooding</i>—Enables/disables syn flooding check.</p> <p>This option protects the Secure Router from syn-flooding attacks.</p> <p>By default, this option is enabled.</p>
	<p><i>tcp-seq-except-bgp-self-port</i>—Enables/disables BGP to use MD5 signatures when <i>tcp-seq-number-predict</i> and <i>tcp-seq-number-range</i> are set. When enabled, it allows any TCP connection with the BGP destination port of 179 to not have the TCP connection resequenced, which causes the MD5 digest to fail.</p> <p>By default, this option is disabled.</p>

Variable	Value
	<p><i>tcp-seq-number-predict</i>—Enable/disables TCP sequence number check. Prevents attempts to predict IP sequence numbers. If an attacker can predict the initial sequence number in the TCP (Transport Control Protocol) handshake, the attacker may be able to hijack the TCP session. This option randomizes the TCP ISNs (Initial Sequence Number) going through the firewall. By default, this option is disabled.</p>
	<p><i>tcp-seq-number-range</i> &lt;20000—2147483647&gt;—Enables/disables TCP sequence number range. An attacker can attempt to replay a captured packet through the firewall by brut-force and thus consume the bandwidth as well as the resources of the target CPU. With this check turned on, the firewall allows only those packets that have sequence numbers in a configured range from the last acknowledgement seen on the connection. The range can be configured with value between 20000 and 2147483647. By default, this option is disabled.</p>
	<p><i>win-nuke</i>—Enables/disables Win-nuke check. The Win-nuke attack sends out-of-band data to an IP address of a Windows machine connected to a network and/or Internet. By default, this option is disabled.</p>

## Configuring global NAT hairpinning

If you enable hairpinning, you disable self-IP connections. If you disable hairpinning, self-IP connections are allowed.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`
3. To enable or disable hairpinning, enter:  
`[no] hairpinning-SelfIp`

**Table 24: Variable definitions**

Variable	Value
[no]	Disables hairpinning.



---

## Configuring peer-to-peer RTP media

Configure Peer-Peer RTP media between clients in a trusted network.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`
3. To specify global ALG configuration, enter:  
`algs`
4. To enable or disable the sip-p2p-media, enter:  
`[no] sip-p2p-media`

**Table 25: Variable definitions**

Variable	Value
[no]	Disables SIP peer-to-peer media.

---

## Configuring self firewall policy with NAT

Configure a self firewall policy with NAT.

Self-NAT allows the traffic generated from routers to be translated from a private IP address to a public IP address. You can use self-NAT to bind the Media Gateway and SSM to private IP addresses and handle the SIP ALG translation using a forward-NAT scenario.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To create the untrusted firewall zone, enter:  
`firewall internet`
3. To configure a self policy with nat, enter:  
`policy 10 out self nat-ip <untrusted-interface>`

**Table 26: Variable definitions**

Variable	Value
[no]	Disables ??

---

## Configuring global IP reassembly

### Enabling and disabling global IP reassembly

Enable or disable IP packet reassembly. By default, IP reassembly is enabled.

IP allows packets to be split in transit and reassembled on delivery. This allows longer packets to be routed through intermediate networks that have rules limiting packets to lengths smaller than the routed packet. The oversized packet can be broken up into pieces small enough to pass. The IP reassembly feature allow the split packets to be reassembled upon delivery.

#### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`
3. To specify global IP reassembly configuration, enter:  
`ip-reassembly`
4. To enable or disable global IP reassembly, enter:  
`[no] enable`

### Configuring global IP reassembly fragment count

Specify the IP reassembly fragment count to control the maximum number of fragments allowed per IP packet. This value limits the number of fragments into which a packet can be fragmented.

#### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify global firewall configuration, enter:  
`firewall global`

3. To specify global IP reassembly configuration, enter:

```
ip-reassembly
```

4. To configure fragment count, enter:

```
fragment-count <1 - 214748364>
```

**Table 27: Variable definitions**

Variable	Value
<1 - 214748364>	Specifies the maximum number of fragments allowed per IP packet. Default value: 44.

## Configuring global IP reassembly fragment size

Set the IP reassembly fragment size to specify the maximum allowable fragment size of the IP packet.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify global IP reassembly configuration, enter:

```
ip-reassembly
```

4. To configure fragment size, enter:

```
fragment-size <1 - 65535>
```

**Table 28: Variable definitions**

Variable	Value
<1 - 65535>	Fragment header length. Default value: 28.

## Configuring global IP reassembly packet-size

Set the maximum size of the IP packet for IP reassembly.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify global IP reassembly configuration, enter:

```
ip-reassembly
```

4. To configure packet size, enter:

```
packet-size <1 - 65535>
```

**Table 29: Variable definitions**

Variable	Value
<1 - 65535>	Specifies the size of the IP packet for reassembly. Default value: 65535.

## Configuring global IP reassembly timeout

Set the IP reassembly timeout value. If a fragmented packet is not reassembled within this time limit, the packet is discarded.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify global IP reassembly configuration, enter:

```
ip-reassembly
```

4. To configure timeout, enter:

```
timeout <11-120>
```

**Table 30: Variable definitions**

Variable	Value
<11-120>	Time value in seconds (default: 60)

---

## Configuring global logging

### Configuring the logging threshold for attacks

Configure a threshold for logging attacks to control the recording of attack logs. Whenever the number of attacks reaches the configured threshold value, a log message is generated. By default, logged information is written to console. If a syslog server is configured, the log information is directed to the syslog server and the console.

#### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify global firewall configuration, enter:  

```
firewall global
```
3. To specify global logging configuration, enter:  

```
logging
```
4. To configure attack logging, enter:  

```
attacks <1-2147483647>
```

**Table 31: Variable definitions**

Variable	Value
<1-2147483647>	Number of attacks logging events (default:100)

### Configuring the logging threshold for an access policy

Configure the access policy logging threshold to control the recording of policy logs. The threshold for policy-based logging defines the number of events against an access policy that are required to generate a log message. By default, logged information is written to the console. If a syslog server is configured, the log information is directed to the syslog server and the console.

#### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify global firewall configuration, enter:  

```
firewall global
```

3. To specify global logging configuration, enter:

```
logging
```

4. To configure policy logging, enter:

```
policy <1-2147483647>
```

**Table 32: Variable definitions**

Variable	Value
<1-2147483647>	Specifies the events threshold for policy logging. Default value: 1.

## Configuring the logging threshold for VPN

Configure the VPN logging threshold to control the recording of VPN logs. When the VPN log reaches the configured threshold value (default is 100 events), a log is created. By default, logged information is written to console. If a syslog server is configured, the log information is directed to the syslog server and the console.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify global logging configuration, enter:

```
logging
```

4. To configure VPN logging, enter:

```
vpn <1-2147483647>
```

**Table 33: Variable definitions**

Variable	Value
<1-2147483647>	Specifies the events threshold for VPN logging. Default value:100.

## Configuring global maximum connection limits for the firewall

Configure the maximum number of allowed connections through the firewall.

## Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure maximum connection limits, enter:

```
max-connection-limit { self | int2self | <map-name> }
<1-29912>
```

**Table 34: Variable definitions**

Variable	Value
<1-29912>	Specifies the number of allowed connections.
self	Specifies maximum number of global self connections.
int2self	Specifies maximum number of global connections from internet to self.
<map-name>	Specifies the name of a map to configure with maximum allowed connections.

---

## Configuring NAT ACL

Use the following procedure to manually configure a NAT ACL.

### Procedure steps

1. To configure NAT ACL, enter Configuration Mode.

```
configure terminal
```

2. Enter the **ip nat** subtree.

```
ip nat
```

3. Create an access list.

```
access-list <listname>
```

4. If applicable, specify an address or range to permit.

```
add permit ip <range-start> <range-end>
```

5. If applicable, specify an address or range to deny.

```
add deny ip <range-start> <range-end>
```

6. Exit the access-list configuration to finish or create another.

```
exit
```

7. Create an address pool.

```
pool <poolname>
```

8. Specify the address pool range. You can specify more than one range using the same command syntax.

```
range <range-start> <range-end> <mask>
```

9. Exit the address pool configuration.

```
exit
```

10. Configure an access group to use the address pool.

```
access-group <groupname> address-pool <poolname>
```

11. If applicable, configure ACL access to a specific NAT module.

```
access-group <groupname> {static | dynamic | address}
```

**Table 35: Variable definitions**

Variable	Value
<groupname>	Specifies the name of access group.
<listname>	Specifies the name of the Access Control List.
<mask>	Specifies the subnet mask of a supplied address range.
<poolname>	Specifies the identifying name of address pool.
<range-end>	Specifies the range end address used when configuring an ACL.
<range-start>	Specifies the address to add or range-start address used when configuring an ACL.
{static   dynamic   address}	Specifies the NAT module to which the ACL applies—static translation, dynamic port translation, or dynamic address translation.

## Configuring NAT failover

Configure NAT failover to allow a primary interface (for example, T1 WAN bundle) using PAT to failover to a backup interface (for example, PPPoE or ISDN). In this case, when the primary interface is up, packets going out through the interface are translated using the IP address of the primary interface. When the primary interface goes down, the IP address of the backup interface is used for the translations, and the stale firewall connections are flushed.

### Procedure steps

1. To enter configuration mode, enter:



```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure NAT failover, enter:

```
nat-failover <primary-interface-name> <secondary-interface-name>
```

---

## Configuring proxy NAT

Enable Proxy NAT to allow SIP trunking to function behind a NAT. With Proxy NAT enabled, the SIP ALG can perform multiple translations within a single packet. The ALG performs a Static NAT translation for the SIP header, and a NAPT translation for the SIP message body (SDP). This results in a single firewall connection between the two call servers on port 5060, for all SIP signaling, and multiple RTP connections for media traffic between the phones.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure proxy NAT, enter:

```
proxy-nat <A.B.C.D>
```

---

## Configuring RFC 3947 NAT traversal acceptance

Use this procedure to configure the SR 2330/4134 to accept or reject an IKE proposal that supports the RFC 3947 version of NAT traversal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To select the IKE policy, enter:

```
ike policy <policy-name> <peer>
```

4. To configure RFC 3947 NAT traversal acceptance, enter:

```
[no] enable-natt-rfc3947
```

**Table 36: Variable definitions**

Variable	Value
[no]	Rejects an IKE proposal that supports the RFC 3947 version of NAT traversal.
<peer>	Specifies the peer IP address or the domain name for IKE negotiations.
<policy-name>	Specifies the IKE policy name, to a maximum of 8 characters.

---

## Configuring global timeout

### Configuring general timeouts

Set the default timeout values for protocols like TCP, UDP, ICMP, FTP and DNS.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify timeout configuration, enter:

```
timeout
```

4. To configure general timers , enter:

```
general {tcp | udp | tcp-reset | icmp | ftp-inactivity | dns-  
inactivity} <0-65535>
```

**Table 37: Variable definitions**

Variable	Value
0-65535	Specifies the timeout in seconds.
{tcp   udp   tcp-reset   icmp   ftp-inactivity   dns-inactivity}	tcp: Specifies the Transport Control Protocol timeout. udp: Specifies the User Datagram Protocol timeout. tcp-reset: Specifies the Transport Control Protocol reset timeout. icmp: Specifies the Internet Control Message Protocol timeout. ftp-inactivity: Specifies how long the File Transport Protocol waits for a response before timing out. dns-inactivity:

Variable	Value
	Specifies how long the Domain Name Service waits for a response before timeout.

## Configuring service record timer

Configures a timeout for a service record.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify timeout configuration, enter:

```
timeout
```

4. To configure service record timers , enter:

```
service <service-name> {tcp | udp} <port-number> <0-65535>
```

**Table 38: Variable definitions**

Variable	Value
service-name	Specifies the name of the service.
0-65535	Specifies the timeout in seconds.
{tcp   udp }	tcp: TCP timeout udp: UDP timeout
port-number	Specifies the TCP or UDP port number.

## Configuring global URL key filters

Configure global URL key filters to filter web access for out bound connections, based on the key words. You can filter up to 20 key words at a time.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure global URL key filters, enter:

```
[no] url-key-filter <key-names>
```

**Table 39: Variable definitions**

Variable	Value
<key-names>	The list of word strings, separated by a space. The maximum list size is 20 words.
[no]	Removes the specified key word string.

## Configuring port trigger records

Configure a port trigger record.

Port triggering lets you define an application-specific customized firewall policy. This feature lets you configure forward or reverse conduits through the firewall that are opened when the trigger application is launched.

Port triggering can be understood as a manual ALG. If an application that is not currently supported by an ALG needs to open port x to function, port-triggering allows you to open the desired port so long as the application is running.

The trigger application is defined by the trigger IP address, protocol and port, and the conduit is defined by a combination of transport protocol and port number.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To configure the port-trigger records, enter:

```
[no] port-trigger <record-name>
```

```
{port <start-port> <end-port>}
```

```
[protocol {tcp | udp}]
```

```
[address <src-ip>]
```

```
{ {forward-direction | reverse-direction} {tcp | udp} <start-port> <end-port>}
```

```
[timeout <timeout>]
```

```
[status {enable | disable}]
```

**Table 40: Variable definitions**

Variable	Value
<record-name>	Port trigger record name.
port {<start-port> <end-port>}	Port trigger ports.
[protocol {tcp   udp}]	Port trigger protocol. Default is TCP.
[address <src-ip>]	Source IP address for port-trigger. Enter a valid IP address, or any. Default is any.
{ {forward-direction   reverse-direction} {tcp   udp} <start-port> <end-port>}	Specifies the protocol (TCP or UDP) and the port numbers (start and end port) to open in the same direction (forward-direction) or opposite direction (reverse-direction) as the established control connection.
[timeout <timeout>]	Port trigger timeout. Default: 600
[status {enable   disable}]	Enables or disables same-direction or reverse-direction conduits through the firewall. (default: enable)

---

## Configuring policy-specific properties

---

### Configuring firewall objects

With firewall objects, you can assign a set of policy parameters to an object entity, and then apply these configured parameters to one or more policies. The ability to differentiate and name different sets of parameters can make your policies easier to write and, when applied, understand.

You can configure objects globally or for a specific firewall zone. When configured globally, configured objects can be applied to any number of policies in any map. When configured for a specific map, the object is available for that map only.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the map name to configure, or global firewall configuration, enter:

```
firewall [global | <map-name>]
```

3. To specify firewall objects configuration, enter:

```
object
```

4. To configure the firewall objects, enter:

```
[no]
```

```
[address <object-name> <ipaddress>]
```

```
[ftp-filter <object-name> {permit | deny | log} <ftp-  
commands>]
```

```
[http-filter <object-name> {deny | log} <web-extensions>]
```

```
[nat-pool <object-name> {static | dynamic | pat} <NAT-  
startip> <NAT-endip>]
```

```
[rpc-filter <object-name> {permit | deny | log} <rpc-  
numbers>]
```

```
[schedule <object-name> [week-day <start-day> <end-day>]
```

```
[start-time <hour> <minutes>] [end-time <hour> <minutes>]
```

```
[service <object-name> {tcp | udp} <port>]
```

```
[smtp-filter <object-name> {permit | deny | log} <smtp-  
commands>]
```

**Table 41: Variable definitions**

Variable	Value
[address <object-name> <ipaddress>]	IP address object. Can be specified as <start-address> <end-address> or <address> <prefix-len>
[ftp-filter <object-name> {permit   deny   log} <ftp-commands>]	List of FTP commands to permit (for example: put, get, ls, mkdir, cd, pasv). Creates an FTP application filter object.
[http-filter <object-name> {deny   log} <web-extensions>]	List of web extensions to deny (for example, java,activex, jar, *.url extension)
[nat-pool <object-name> {static   dynamic   pat} <NAT-startip> <NAT-endip>]	Specifies the start IP and end IP for NAT, in the form A.B.C.D or by specifying the address object name.
[rpc-filter <object-name> {permit   deny   log} <rpc-numbers>]	List of RPC numbers to permit or deny. The RPC allow filter allows only the listed program numbers and denies the others. An RPC deny filter denies only the listed programs and allows the others.
[week-day <start-day> <end-day>]	Specifies the days of the week for the schedule. Can be sun, mon, tue, wed, thu, fri, or sat. For example mon tue.
[start-time <hour> <minutes>]	Activation time on each specified day.

Variable	Value
[end-time <hour> <minutes>]	Deactivation time on each specified day.
[service <object-name> {tcp   udp} <port>]	TCP or UDP port. Can be specified as <start-port> <end-port> or <port>.
[smtp-filter <object-name> {permit   deny   log} <smtp-commands>]	List of SMTP commands to permit or deny (for example, helo, mail, rcpt, data, quit, send, saml, rset, verify, expn).

## Configuring connection reservations

Configure connection reservations.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the firewall map to configure, enter:

```
firewall <map-name>
```

3. To configure global connection reservations, enter:

```
connection-reservation {out|in} <1-29912> {<address> |  
<start-address> <end-address> | <address-object> }
```

**Table 42: Variable definitions**

Variable	Value
{out   in}	out: outbound direction in: inbound direction
<1-29912>	Number of connections to reserve.
{<address>   <start-address> <end-address>   <address-object>}	Specifies the destination address for inbound traffic from the Internet, or the source address for traffic that is outbound from the trusted zone.

## Configuring reset of invalid ACK packets

Enable the sending of a reset when the firewall detects an invalid ACK packet. You can configure this feature globally or for a specific firewall zone.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the map name to configure, or global firewall configuration, enter:

```
firewall {global | <map-name>}
```

3. To configure reset of invalid ACK packets, enter:

```
[no] reset-invalid-acks
```

**Table 43: Variable definitions**

Variable	Value
[no]	Disables reset-invalid-acks.

---

## Configuring stealth mode

Enable stealth mode on the firewall to stop the sending of TCP reset packets when there is no corresponding matching policy for an incoming packet. You can configure this option either globally or for a specific firewall zone.

By default, this feature is disabled.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the map name to configure, or global firewall configuration, enter:

```
firewall {global | <map-name>}
```

3. To configure reset of invalid ACK packets, enter:

```
[no] stealth-mode [<map-name>]
```

**Table 44: Variable definitions**

Variable	Value
<map-name>	Specifies the map name.
[no]	Disables stealth mode.

---

## Configuring firewall policies

Configure firewall policies for a specific map. The maximum number of policies for each map is 1024.



## Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the map name to configure, enter:  

```
firewall <map-name>
```
3. To configure policies for the firewall, enter:  

```
[no] policy <1-1024> {out|in}
[action {permit | deny | reject}]
[address <ipaddress>]
[service <service-name>]
[protocol <protocol-value>]
[port <port-value>]
[traffic {transit | self}]
[nat-ip <nat-ip-value>]
[port-map <port-map-value>]
[log {enable-log | disable-log}]
```

**Table 45: Variable definitions**

Variable	Value
<1-1024>	Specifies the map priority, and also uniquely identifies the map.
{out in}	Specifies the traffic direction in which the policy is applied.
[action {permit   deny   reject}]	Specifies the action of the policy. If none is specified, permit is the default action.
[address <ipaddress>]	IP address. Can be specified as <src-ip> <prefix-len> <dst-ip> <prefix-len> OR <src-start> <src-end> <dst-start> <dst-end> OR <src-object> <dst-object>
[service <service-name>]	Service name, or <b>any</b> to specify any service.
[protocol <protocol-value>]	Can be one of the following: tcp udp icmp ah esp gre any
[port <port-value>]	Can be specified as <src-start> <src-end> <dst-start> <dst-end> OR <src-port> <dst-port>
[traffic {transit   self}]	Type of traffic, either transit or self. Default is transit.
[nat-ip <nat-ip-value>]	Specifies the IP address or interface to use for NAT translation. Can be ethernet<slot/port>, intf-name, intf-name:pvc-num

Variable	Value
[port-map <port-map-value>]	Port to application mapping (pam) for reverse NAT IP only. Can be specified as <start-port> <end-port> or <port>.
[log {enable-log   disable-log}]	Enable or disable logging.
[no]	Deletes the specified policy from the map.

---

## Applying an object to a policy

Apply pre-configured objects to a policy.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the map name to configure, enter:  
`firewall <map-name>`
3. To specify the firewall policy to configure, enter:  
`policy <1-1024> {out|in}`
4. To apply an object to the policy, enter:  
`apply-object <object-type> <object-name>`

**Table 46: Variable definitions**

Variable	Value
<object-type>	Specifies the object type. Valid options are: ftp-filter http-filter smtp-filter rpc-filter schedule nat-pool
<object-name>	Specifies the object name.

---

## Configuring bandwidth for the policy

Specify the maximum allowed bandwidth for the policy in kilobytes per second. By default, this feature is disabled.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the map name to configure, enter:

```
firewall <map-name>
```

3. To specify the firewall policy to configure, enter:

```
policy <1-1024> {out|in}
```

4. To specify the bandwidth for the policy in kilobytes per second, enter:

```
[no] bandwidth <1-4194303>
```

**Table 47: Variable definitions**

Variable	Value
<1-4194303>	Specifies the maximum bandwidth in kilobytes per second.
[no]	Disables the bandwidth-limiting feature for this policy (disabled by default).

## Configuring the maximum connections for the policy within a configured timeframe

Specify the maximum number of connections for a given policy in a particular time.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the map name to configure, enter:

```
firewall <map-name>
```

3. To specify the firewall policy to configure, enter:

```
policy <1-1024> {out|in}
```

4. To configure the maximum concurrent connections for the policy, enter:

```
[no] connection-rate <1-38160> [<sample-time>]
```

**Table 48: Variable definitions**

Variable	Value
<1-38160>	Specifies the maximum number of connections.
[no]	Disables the feature.
[<sample-time>]	Specifies the sample time in seconds. Valid range is 1-36000. If not specified, the default value is used (1 second).

---

## Configuring the maximum connections for the policy

Specify the maximum number of connections for a given policy at any given time.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the map name to configure, enter:  

```
firewall <map-name>
```
3. To specify the firewall policy to configure, enter:  

```
policy <1-1024> {out|in}
```
4. To configure the maximum number of connections for the policy, enter:  

```
[no] max-connection-limit <1-29912>
```

**Table 49: Variable definitions**

Variable	Value
<1-29912>	Specifies the maximum number of connections for the policy.
no	Resets the number of connections to the default value, which is the maximum number of connections for the current map.

---

## Configuring policing for the policy

Configure policing to control the maximum flow rate for a given policy in packets per second. By default, this feature is disabled.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the map name to configure, enter:  

```
firewall <map-name>
```
3. To specify the firewall policy to configure, enter:  

```
policy <1-1024> {out|in}
```
4. To configure policing for the policy, in packets per second, enter:

```
[no] policing <1-2147483647>
```

**Table 50: Variable definitions**

Variable	Value
<1-2147483647>	Specifies the maximum number of packets per second.
[no]	Disables the policing feature. (By default, this feature is disabled.)

---

## Enabling the policy

Enable or disable the policy. By default, the policy is enabled.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the map name to configure, enter:  

```
firewall <map-name>
```
3. To specify the firewall policy to configure, enter:  

```
policy <1-1024> {out|in}
```
4. To enable or disable the policy, enter:  

```
[no] enable
```

**Table 51: Variable definitions**

Variable	Value
[no]	Disables the policy.

---

## Adding interfaces to the firewall zone

Add one or more interfaces to a map. Up to 32 interfaces can be added to one zone, with a maximum of five interfaces specified at one time.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the firewall map to configure, enter:

```
firewall <map-name>
```

3. To add an interface to the zone, enter:

```
[no] interface <interface-name>
```

**Table 52: Variable definitions**

Variable	Value
<interface-name>	The interface can be specified as one of the following: ethernet<slot/port> <bundle-name> <bundle-name>:<pvc-number> Up to five interfaces, separated by spaces, can be listed at once.
[no]	Removes the specified interfaces from the zone.

---

## Displaying firewall information

To display firewall configuration information, enter:

```
show firewall
```

```
algs
```

```
attack-checks
```

```
bypass-trusted
```

```
connection-reservation <map-name> [statistics]
```

```
connections <map-name> [address <A.B.C.D>] [port <port>]  
[protocol <protocol>] [summary]
```

```
hairpinning
```

```
interface <map-name>
```

```
ip-reassembly
```

```
logging
```

```
maps
```

```
max-connection-limit <map-name>
```

```
nat-failover
```

```
nat-translations <map-name> [address <A.B.C.D>] [port <port>]  
[protocol <protocol>]
```

```
object <object-type> {<map-name> | global} <object-name>
```

```
policy <map-name> [priority <1-1024>] [statistics] [detail]  
port-trigger <port-trigger-name>  
proxy-nat  
reset-invalid-acks  
statistics  
stealth-mode  
timeout [general | <service-name>]  
url-key-filter
```

---

## Clearing firewall connections

To clear firewall connections, enter:

```
clear firewall connection {<ip-address>|all}
```

---

## Clearing firewall statistics

To clear firewall statistics, enter:

```
clear firewall statistics
```





# Chapter 12: Packet filter configuration

This chapter describes how to perform packet filter configurations.

---

## Configuring IPv4 packet filters

Configure IPv4 packet filters to be applied to one or more interfaces.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the IP packet filter to create or configure, enter:  
`ip packet-filter <packet-filter-name>`
3. To configure rules for the packet filter, enter:  
`[no] {add | insert <lineno>}`  
`{permit | deny}`  
`{tcp | udp | icmp | ip | igmp | <0-255>}`  
`<src-address> <dst-address>`  
`[sport <src-port>]`  
`[dport <dst-port>]`  
`[icmptype <icmp-type>]`  
`[icmpcode <icmp-code>]`  
`[igmpdtype <igmp-type>]`  
`[precedence <precedence>]`  
`[dscp <dscp>]`  
`[tos <tos>]`  
`[flags <tcp-flags>]`  
`[fragments {on|off}]`  
`[log {on|off}]`

```
[expire <expiry-time>]
```

**Table 53: Variable definitions**

Variable	Value
{permit   deny}	Specifies the action to perform when a packet matches the filter rule. permit: allow the packet to cross the filter deny: drop the packet When a packet matches a filter rule, no further packet filter list processing is performed.
{tcp   udp   icmp   ip   igmp   <0-255>}	Specifies the name or number of an Internet protocol. This can be one of the key words (TCP, UDP, ICMP, IGMP, or IP), or an integer in the range 0 - 255 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword IP. Transit network packets must include a matching protocol to match this packet filter rule.
<src-address>	Specifies the source host or network IP address, subnet, or wildcard. <A.B.C.D>/0-32: matches a subnet based on the prefix. any: matches any source IP address. <A.B.C.D>/<A.B.C.D>: matches a source IP address based on the wildcard. The wildcard is a dotted four digit number that follows the forward slash (/). A wildcard is considered an anti-mask. In a wildcard, 0 is the match-exactly bit and a 1 is the match anything bit. (For example, 1.2.3.4/0.0.0.255 is the same as 1.2.3.0/24.) Wildcards can also express match criteria that is noncontiguous. (For example, 1.2.3.4/255.0.255.0 will match any packet from 1.*.2.*. ) Transit network packets must include a matching source IP address to match this packet filter rule.
<dst-address>	Specifies the destination host or network IP address, subnet, or wildcard. <A.B.C.D>/0-32: matches a subnet based on the prefix. any: matches any source IP address. <A.B.C.D>/<A.B.C.D>: matches a source IP address based on the wildcard. The wildcard is a dotted four digit number that follows the forward slash (/). A wildcard is considered an anti-mask. In a wildcard, 0 is the match-exactly bit and a 1 is the match anything bit. (For example, 1.2.3.4/0.0.0.255 is the same as 1.2.3.0/24.) Wildcards can also express match criteria that is noncontiguous. (For example, 1.2.3.4/255.0.255.0 will match any packet from 1.*.2.*. ) Transit network packets must include a matching destination IP address to match this packet filter rule.
[sport <src-port>] [dport <dst-port>]	Optional entry for TCP and UDP protocols; allows the source or destination port to be filtered. =p: Specifies port number p, where p is 1- 65535. !=p: Excludes port p. >p: Specifies any port number greater than p >=p: Specifies any port number greater than or equal to p <p: Specifies any port number less than p <=p: Specifies any port number less than or equal to

Variable	Value
	<p>p p1-p2: Specifies any port number within the range p1 - p2. Transit network packets must include a matching source or destination TCP or UDP port to match this packet filter rule.</p> <p><b>Note:</b></p> <p>Due to hardware limitations, port range is not supported on Secure Router 2330 yet is supported on Secure Router 4134.</p>
[icmptype <icmp-type>]	An optional parameter that specifies the ICMP message type to be filtered. If the protocol parameter is configured to icmp, this rule will match only ICMP packets of this type. Values ranges from 0 to 255.
[icmpcode <icmp-code>]	An optional parameter that specifies the ICMP message code to be filtered, if specified along with a message type. If the protocol parameter is configured to icmp, this rule will match only ICMP packets with this code. Values ranges from 0 to 255.
[igmp-type <igmp-type>]	Specifies the IGMP type: group-query v1-report dvmrp pim trace v2-report v2-leave mtrace-response mtrace v3-report mra mrs mrt or 0-12
[precedence <precedence>]	Specifies the IP header precedence value to be filtered. Values range from 0 to 7. The Avaya Secure Router 2330/4134 supports configuring precedence and TOS together, but you cannot configure precedence or TOS together with DSCP. Configuring a precedence number requires that the transit network packet include the same number set in the precedence field to match this packet filter rule.
[tos <tos>]	Specifies the IP header type of service (TOS) value to be filtered (for UDP and ICMP protocols). Values range from 0 to 15. The Secure Router 2330/4134 supports configuring TOS and precedence together, but you cannot configure precedence or TOS together with DSCP.
[dscp <dscp>]	Specifies the IP differentiated services code point (dscp). Values range from 0 to 63. You cannot configure DSCP together with precedence or TOS. Transit network packets must include a matching number in the DS field to match this packet filter rule.
[flags <tcp-flags>]	Specifies the TCP flags to be filtered. You can specify multiple TCP flags by separating the keywords below with commas (no spaces allowed). This entry may be any of the following words: established: Used to match an established connection (Cisco-compatible). fin: Matches the TCP FIN header flag. syn: Matches the TCP SYN header flag. ack: Matches the TCP ACK header flag. psh: Matches the TCP

Variable	Value
	PSH header flag. rst: Matches the TCP RST header flag. urg: Matches the TCP URG header flag. If one or more flags is configured, the transit network packet must include these TCP flags to match this packet filter rule. If a list of flags is selected, the transit network packet must include all flags in the list to match this packet filter rule.
[fragments {on off}]	Matches non-initial fragmented packets where the fragment extension header contains a non-zero fragment offset. The fragments keyword is an option only if the operator [port-number] arguments are not specified. on: requires the transit network packet be a non-initial fragment off: requires the transit network packet be an initial fragment or unfragmented
[log {on off}]	Specifies whether or not a logging message is reported to the user when a rule match occurs. All packets matching this rule will be logged at the debug priority level. Cache flow creation and expiration events will be logged at the information priority level. A periodic report every 5 minutes with the number of packets matching a particular cache flow will be logged at the notice priority level. on: Turns on logging. off: Turns off logging. (default) You must also enable either console or syslogging in the system logging subtree.
[expire <expiry-time>]	Specifies the rule expiry time in seconds. Expired rules disappear from the configuration file after the expiry timeout. Using the expire parameter can be helpful when network administrators are concerned a network is under temporary attack or they want to temporarily debug what causes a network to block certain traffic.
[no]	Removes the specified packet filter.

---

## Configuring IPv6 packet filters

Configure IPv6 packet filters to be applied to one or more interfaces.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the name of the IPv6 packet filter to create or configure, enter:  

```
ipv6 packet-filter <packet-filter-name>
```
3. To configure rules for the packet filter, enter:

```

[no] {add | insert <lineno>}
{permit | deny}
{tcp | udp | icmp | ipv6 | <0-255>}
<src-address> <dst-address>
[sport <src-port>]
[dport <dst-port>]
[icmptype <icmp-type>]
[icmpcode <icmp-code>]
[dscp <dscp-value>]
[flowlabel <flowlabel-value>]
[flags <tcp-flags>]
[routing {on|off}]
[log {on|off}]
[expire <expiry-time>]

```

**Table 54: Variable definitions**

Variable	Value
{permit   deny}	Specifies the action to perform when a packet matches the filter rule. permit: allow the packet to cross the filter deny: drop the packet When a packet matches a filter rule, no further packet filter list processing is performed.
{tcp   udp   icmp   ip   igmp   <0-255>}	Specifies the name or number of an Internet protocol. This can be one of the key words (TCP, UDP, ICMP, IGMP, or IP), or an integer in the range 0 - 255 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword IP. Transit network packets must include a matching protocol to match this packet filter rule.
<src-address>	Specifies the source host or network IPv6 address, subnet, or wildcard. <A.B.C.D>: matches a specific IPv6 address. <A.B.C.D>/0-128: matches an ipv6 subnet with the CIDR subnet bit mask expressed as the 0-128 behind the forward slash (/). any: matches any source IPv6 address. Transit network packets must include a matching source IPv6 address to match this packet filter rule.
<dst-address>	Specifies the destination host or network IPv6 address, subnet, or wildcard. <A.B.C.D>: matches a specific IPv6 address. <A.B.C.D>/0-128: matches an ipv6 subnet with the CIDR subnet bit mask expressed as the 0-128 behind the

Variable	Value
	forward slash (/). any: matches any source IPv6 address. Transit network packets must include a matching destination IPv6 address to match this packet filter rule.
[sport <src-port>] [dport <dst-port>]	Optional entry for TCP and UDP protocols; allows the source or destination port to be filtered. =p: Specifies port number p, where p is 1- 65535. !=p: Excludes port p. >p: Specifies any port number greater than p >=p: Specifies any port number greater than or equal to p <p: Specifies any port number less than p <=p: Specifies any port number less than or equal to p p1-p2: Specifies any port number within the range p1 - p2 Transit network packets must include a matching source or destination TCP or UDP port to match this packet filter rule.  <b>Note:</b> Due to hardware limitations, port range is not supported on Secure Router 2330 yet is supported on Secure Router 4134.
[icmptype <icmp-type>]	An optional parameter that specifies the ICMP message type to be filtered. If the protocol parameter is configured to icmp, this rule will match only ICMP packets of this type. Values ranges from 0 to 255.
[icmpcode <icmp-code>]	An optional parameter that specifies the ICMP message code to be filtered, if specified along with a message type. If the protocol parameter is configured to icmp, this rule will match only ICMP packets with this code. Values ranges from 0 to 255.
[igmptype <igmp-type>]	Specifies the IGMP type: group-query v1-report dvmrp pim trace v2-report v2-leave mtrace-response mtrace v3-report mra mrs mrt or 0-12
[dscp <dscp-value>]	IP DSCP value. Range is 0-63. Transit network packets must include a matching number in the DS field to match this packet filter rule.
[flowlabel <flowlabel-value>]	Flow label value. Range is 0-1048575. Transit network packets must include a matching number to match this packet filter rule.
[flags <tcp-flags>]	Specifies the TCP flags to be filtered. You can specify multiple TCP flags by separating the keywords below with commas (no spaces allowed). This entry may be any of the following words: established: Used to match an established connection (Cisco-compatible). fin: Matches the TCP FIN header flag. syn: Matches the TCP SYN header flag. ack: Matches the TCP ACK header flag. psh: Matches the TCP PSH header flag. rst: Matches the TCP RST header flag. urg: Matches the TCP URG header flag. If one or more flags is configured, the transit network packet must include these

Variable	Value
	TCP flags to match this packet filter rule. If a list of flags is selected, the transit network packet must include all flags in the list to match this packet filter rule.
[routing {on off}]	Specifies whether filter of routing header is on or off. Transit network packets must include a routing header to match this packet filter rule.
[log {on off}]	Specifies whether or not a logging message is reported to the user when a rule match occurs. All packets matching this rule will be logged at the debug priority level. Cache flow creation and expiration events will be logged at the information priority level. A periodic report every 5 minutes with the number of packets matching a particular cache flow will be logged at the notice priority level. on: Turns on logging. off: Turns off logging. (default) You must also enable either console or syslogging in the system logging subtree.
[expire <expiry-time>]	Specifies the rule expiry time in seconds. Expired rules disappear from the configuration file after the expiry timeout. Using the expire parameter can be helpful when network administrators are concerned a network is under temporary attack or they want to temporarily debug what causes a network to block certain traffic.
[no]	Removes the specified packet filter.

---

## Configuring MAC packet filters

Configure MAC packet filters to be applied to one or more interfaces.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the MAC packet filter to create or configure, enter:

```
mac packet-filter <packet-filter-name>
```

3. To configure rules for the packet filter, enter:

```
[no] {add <lineno> | insert <lineno>}
{permit | deny}
[src-mac <any | a.b.c>]
[dst-mac <any | a.b.c>]
```

```
[smask <src-mask>] [dmask <dst-mask>]
[ethertype <ether-type>]
[cos <cos-value>]
[vlan <vlan-id>]
```

**Table 55: Variable definitions**

Variable	Value
{permit   deny}	Specifies the action to perform when a packet matches the filter rule: permit: allow the packet to cross the filter deny: drop the packet When a packet matches a filter rule, no further packet filter list processing is performed.
[src-mac <any   a.b.c>]	Source MAC address. any: matches any MAC source address a.b.c: matches a specific MAC source address Transit network packets must have a matching source MAC address to match this packet filter rule.
[dst-mac <any   a.b.c>]	Destination MAC address. any: matches any MAC destination address a.b.c: matches a specific MAC destination address Transit network packets must have a matching destination MAC address to match this packet filter rule.
[smask <src-mask>]	Specifies MAC source mask. When you configure a specific source MAC address (a.b.c), you can use the smask parameter to identify a group of MAC addresses that can match. This parameter is expressed as a series of three four digit hexadecimal numbers separated by a period (for example: 1111.2222.0000). It is common to use the letters O and F, but this is not required. For example, the mask 0.0.ffff will match any MAC address with the 4 bytes to the extreme left matching regardless of difference in the 2 bytes to the extreme right.
[dmask <dst-mask>]	Specifies MAC destination mask. When you configure a specific destination MAC address (a.b.c), you can use the dmask parameter to identify a group of MAC addresses that can match. This parameter is expressed as a series of three four digit hexadecimal numbers separated by a period (for example: 1111.2222.0000). It is common to use the letters O and F, but this is not required. For example, the mask 0.0.ffff will match any MAC address with the 4 bytes to the extreme left matching regardless of difference in the 2 bytes to the extreme right.
[ethertype <ether-type>]	Specifies the Ethernet type: arp, mpls, aarp, ppp or a four digit hexadecimal number. To match this rule, the ethertype field in the transit network packet must match the value configured for this ethertype parameter.



Variable	Value
[cos <cos-value>]	Specifies Class of Service (CoS). Values range from 0 to 7. When this parameter is configured, the transit network packet must include a VLAN header and a matching CoS to match this packet filter rule.
[vlan <vlan-id>]	Specifies the VLAN ID. The value is a number ranging from 0 to 4095. When this parameter is configured, the transit network packet must include a VLAN header and a matching VLAN ID to match this packet filter rule.
[no]	Removes the specified packet filter.

---

## Applying a packet filter to an interface

Apply a packet filter to an interface. With WAN modules and chassis Ethernet ports, you can apply one IPv4 and one IPv6 packet filter to an interface in either direction (you cannot enable the IPv4 packet filter and the firewall on the same interface). With Ethernet module interfaces, you can apply one IPv4, one IPv6, and one MAC packet filter in the inbound direction only (no restrictions related to firewalls apply).

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. Apply an existing packet filter rule set to the interface, enter:

```
packet-filter-group <interface-name> [in | out] {[mac <mac-  
packet-filter>] | [ip <ip-packet-filter>] | [ipv6 <ipv6-  
packet-filter>]}
```

---

## Applying a management services packet filter

Use this procedure to apply a management service packet filter globally on a Secure Router.

### Prerequisites

- Configure an IPv4 or IPv6 packet filter.
- Configure a rule list for the packet filter.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To apply a packet filter globally, enter:

```
[no] packet-filter-group management [in | out] [ip <packet-  
filter-name> | ipv6 <packet-filter-name>]
```

**Table 56: Variable definitions**

Variable	Value
[in   out]	Applies the management service packet filter to inbound (in) or local outbound (out) packets. You can apply an individual management service packet filter to both inbound and outbound traffic.
ip <packet-filter-name>	Specifies the IPv4 management service packet filter to apply.
ipv6 < packet-filter-name>	Specifies an IPv6 management service packet filter to apply.

### Job aid: sample configuration

This section provides a sample configuration for creating an IPv4 packet filter, configuring a rule list for the packet filter, and applying the packet filter globally for system management service.

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure an IPv4 packet filter named **ipservice**, enter:

```
ip packet-filter ipservice
```

3. To configure a rule list for the packet filter, enter:

```
add deny tcp any any dport =23  
add permit tcp any any  
exit
```

4. To apply the packet filter globally to inbound packets, enter:

```
packet-filter-group management in ip ipservice
```

---

## Deleting rules from packet filters

Delete rules from configured packet filters.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the packet filter to create or configure, enter:

```
{mac | ip | ipv6} packet-filter <packet-filter-name>
```

3. To specify the line number of the rules to delete from the packet filter, enter:

```
delete <1-65535>
```

---

## Deleting a packet filter

Delete configured packet filters. To delete a packet filter, you must first remove it from all associated interfaces.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the packet filter to delete, enter:

```
no {mac | ip | ipv6} packet-filter <packet-filter-name>
```

---

## Clearing packet filter counters

Clear packet filter list counters.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To clear IPv4 packet filter list counters, enter:

```
clear ip packet-filter counters {<packet-filter-list-name> |  
<all>}
```

3. To clear IPv6 packet filter list counters, enter:

```
clear ipv6 packet-filter counters {<packet-filter-list-name>  
| <all>}
```

---

## Clearing packet filter statistics

Clear packet filter interface statistics.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To clear IPv4 interface packet filter statistics, enter:

```
clear ip packet-filter statistics {<ethernet> | <bundle-name>  
| <bundle-name:pvc_no> | <all>}
```

3. To clear IPv6 interface packet filter statistics, enter:

```
clear ipv6 packet-filter statistics {<ethernet> | <bundle-  
name> | <all>}
```

---

## Displaying packet filters

### Procedure steps

To display a packet filter, enter:

```
show packet-filter <packet-filter-name>
```

---

## Displaying packet filters applied to an interface

### Procedure steps

To display a packet filter, enter:

```
show packet-filter-rules <interface-name>
```

---

## Displaying global management service packet filter information

Use this procedure to display configuration information for global management service packet filters.

### Procedure steps

To display the packet filter information, enter:

```
show packet-filter-rules management
```

---

## Displaying global IPv4 management service packet filter statistics

Use this procedure to display statistical information for IPv4 management service packet filters.

### Procedure steps

To display IPv4 management service packet filter statistics, enter:

```
show ip packet-filter-stats management
```

---

## Displaying global IPv6 management service packet filter statistics

Use this procedure to display statistical information for IPv6 management service packet filters.

### Procedure steps

To display IPv6 management service packet filter statistics, enter:

```
show ipv6 packet-filter-stats management
```



# Chapter 13: IPsec VPN configuration

This chapter describes how to perform IPsec VPN configurations.

The following are the high-level configuration steps for Site-to-site VPN:

## **Important:**

IPsec is only supported on the Avaya Secure Router 2330/4134 when the VPN/IPsec module is installed on the chassis.

1. Configure at least one trusted interface and one untrusted interface.
2. Configure an IKE policy for a specific remote gateway
3. Configure one or more IPsec policies for the same remote gateway
4. Configure an IP route (specific or default) for the destination addresses specified in the IPsec policies

Even though the application traffic, matching the IPsec policy, is getting tunneled, the built-in firewall uses the IP route to cross check whether the router is expected to handle this traffic at all.

5. Configure an inbound firewall policy in the internet zone for IKE negotiation (UDP 500).
6. If a NAT-in-the-middle exists between the peers, configure an inbound firewall policy in the internet zone for IKE negotiation with NAT traversal (UDP 4500).
7. If you are configuring L2TP remote access VPN, configure an inbound firewall policy in the internet zone for L2TP (UDP 1701).
8. If you are configuring a management tunnel, configure inbound firewall policies in the internet map for the required services (telnet, icmp, and so on)
9. If you are configuring a transit tunnel, configure inbound firewall policies in the appropriate map (for example, corp) for the required services

In order for traffic to be forwarded through the VPN, a static or dynamic route to the peer must be available.

---

## Configuring IKE for site-to-site VPN

---

### Creating an IKE policy

Create an IKE policy for a dynamic ISAKMP SA.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To create the IKE policy for site-to-site VPN, enter:

```
ike policy <policy-name> <peer>
```

**Table 57: Variable definitions**

Variable	Value
<policy-name>	Specifies the IKE policy name. Max 13 characters.
<peer>	Specifies the peer IP address or the domain name for IKE negotiations.

---

### Configuring the local address for IKE negotiations

Configure the local address for IKE negotiations. The local address and the address in the certificate must match.

When executed, this command creates an IKE policy proposal with default values of Preshared Key, 3DES, SHA1, and DH-group2.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```



3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To configure the local address for IKE negotiations, enter:

```
local-address <A.B.C.D>
```

---

## Configuring the IKE policy local ID

Configure the local ID to specify the IPsec identifiers for the host that is used in the identification payload during IKE negotiation.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To configure the local ID, enter:

```
local-id {domain-name <fqdn> | email-id <email> | der-  
encoded-dn <name> | key-id <key_id>}
```

**Table 58: Variable definitions**

Variable	Value
domain-name <fqdn>	Specifies a fully qualified domain name (FQDN), like router.com.
email-id <email>	Specifies a fully-qualified email user name string, like name@router.com.
der-encoded-dn <name>	Specifies the x.500 (LDAP) distinguished name.
key-id <key_id>	Specifies vendor specific information used in aggressive mode.

---

## Configuring the IKE policy remote ID

Configure the remote ID to specify the IPsec peer that participates in the IKE negotiation.

## Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To configure the remote ID, enter:

```
remote-id {domain-name <fqdn> | email-id <email>| der-  
encoded-dn <name> | key-id <key_id>}
```

**Table 59: Variable definitions**

Variable	Value
domain-name <fqdn>	Specifies a fully qualified domain name (FQDN), like router.com.
email-id <email>	Specifies a fully-qualified email user name string, like name@router.com.
der-encoded-dn <name>	Specifies the x.500 (LDAP) distinguished name.
key-id <key_id>	Specifies vendor specific information used in aggressive mode.

## Configuring the IKE mode

Configure the IKE mode for the policy. The default mode is main mode.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To configure the IKE mode, enter:

```
[no] mode {aggressive | main}
```

**Table 60: Variable definitions**

Variable	Value
main	Specifies use of full negotiation to establish a security association. Main mode provides identity protection.
aggressive	Specifies use of quick negotiation to establish a security association. Aggressive mode does not provide identity protection.
[no]	Sets the mode to the default value (main mode).

## Configuring the IKE exchange type

Configure the IKE exchange type to control whether the policy can initiate negotiations, respond to negotiations, or both.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To configure the IKE exchange type, enter:

```
exchange-type {initiator-only | responder-only | both}
```

**Table 61: Variable definitions**

Variable	Value
initiator-only	Specifies that the policy cannot respond to IKE negotiations initiated by another party. This type is not supported if the IKE policy is configured for main mode and Preshared Key is configured as the authentication mode.
responder-only	Specifies that the policy cannot initiate IKE to any other part, but responds to the IKE initiated by others.
both	Specifies that the policy can initiate IKE to the other party and also respond to IKE negotiations initiated by another. This is the default setting.

---

## Configuring the pre-shared key for IKE

Define a pre-shared key for the IKE policy. The key is valid only when the proposal configured has the authentication method as pre-shared-key.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IKE policy to configure, enter:  
`ike policy <policy-name> <peer-address>`
4. To configure the IKE key, enter:  
`key <key-string>`

**Table 62: Variable definitions**

Variable	Value
<key-string>	Specifies the pre-shared key. Max 49 characters.

---

## Configuring key usage extension checking

Use this procedure to configure an IKE policy with key usage extension checking.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IKE policy, enter:  
`ike policy <policy-name> <peer>`
4. To enable or disable key usage extension checking, enter:  
`[no] keyusage`

---

## Enabling or disabling PFS

Enable or disable Perfect Forward Secrecy (PFS) of both keys and identities (RFC 2409) for the IKE policy.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IKE policy to configure, enter:  
`ike policy <policy-name> <peer-address>`
4. To enable or disable PFS, enter:  
`[no] pfs`

---

## Configuring IKE proposal

Configure an IKE proposal for a dynamic ISAKMP SA.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IKE policy to configure, enter:  
`ike policy <policy-name> <peer-address>`
4. To create the IKE proposal, enter:  
`[no] proposal <priority>`

**Table 63: Variable definitions**

Variable	Value
<priority>	Specifies the proposal priority, from 1 to 5.
[no]	Deletes the proposal.

## Configuring authentication method for IKE proposal

Configure the IKE proposal authentication method to authenticate the peers .

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To select the IKE proposal to configure, enter:

```
proposal <priority>
```

5. To configure the IKE authentication method, enter:

```
authentication-method {pre-shared-key | dss-signature | rsa-  
signature}
```

**Table 64: Variable definitions**

Variable	Value
pre-shared-key	Authentication using a pre-shared key, derived out of band.
dss-signature	Authentication using Digital Signature Standard
rsa-signature	Authentication using RSA Signature

## Configuring DH group for IKE proposal

Configure the IKE Diffie-Hellman group for key exchange between the peers. This specifies the type of Diffie-Hellman prime modulus group that IKE uses for the key exchange.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer>
```

4. To select the IKE proposal to configure, enter:

```
proposal <priority>
```

5. To configure DH group for IKE proposal, enter:

```
dh-group {group1 | group2 | group5 group14 | group15}
```

**Table 65: Variable definitions**

Variable	Value
group1	768-bit. RFC 2409.
group2	1024-bit. RFC 2409.
group5	1536-bit. RFC2409. This is the highest level of security and requires more processing time than group 1 and group 2.
group14	2048-bit modular exponential (MODP) group. RFC 2409.
group15	3072-bit MODP group. RFC 2409.

## Configuring encryption algorithm for IKE proposal

Configure encryption algorithm for the IKE proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To select the IKE proposal to configure, enter:

```
proposal <priority>
```

5. To configure the encryption algorithm for IKE, enter:

```
encryption-algorithm {des-cbc | 3des-cbc | aes128-cbc |  
aes192-cbc | aes256-cbc}
```

**Table 66: Variable definitions**

Variable	Value
des-cbc	Specifies DES-CBC encryption.
3des-cbc	Specifies 3DES-CBC encryption.

Variable	Value
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.

## Configuring hash algorithm

Configure the IKE authentication algorithm for a given proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IKE policy to configure, enter:  
`ike policy <policy-name> <peer-address>`
4. To select the IKE proposal to configure, enter:  
`proposal <priority>`
5. To configure the hash algorithm, enter:  
`[no] hash-algorithm {md5|sha1}`

**Table 67: Variable definitions**

Variable	Value
md5	Specifies a 128-bit message digest (RFC 1321).
sha1	Specifies Secure Hash Standard, a 160-bit message digest (NIST,FIPS PUB 180-1).
[no]	Sets the hash algorithm to the default value (sha1).

## Configuring lifetime

Configure the lifetime of the IKE SA. When the SA expires, it is replaced by a new negotiated SA or terminated.

### Procedure steps

1. To enter the configuration mode, enter:



```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To select the IKE proposal to configure, enter:

```
proposal <priority>
```

5. To configure lifetime, enter:

```
lifetime {kilobytes <300-4194303> | seconds <300-864000>}
```

**Table 68: Variable definitions**

Variable	Value
kilobytes <300-4194303>	Specifies the volume of traffic (in kilobytes) that can pass between IPsec peers using the given IKE SA before that SA expires. Default: unlimited.
seconds <300-864000>	Specifies the number of seconds the IKE SA runs before expiring. Default: 86400 seconds (24 hours).

## Configuring OCSP for the IKE policy

Enable OCSP to configure the router to contact the CA for verification of the status of any certificate that the router receives.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IKE policy to configure, enter:

```
ike policy <policy-name> <peer-address>
```

4. To enable or disable OCSP, enter:

```
[no] ocsp
```

## Configuring IPsec for site-to-site VPN

### Creating an IPsec policy

Create an IPsec policy for an IPsec SA.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To create the IPsec policy, enter:

```
ipsec policy <policy-name> <peer> [before-name <name>]  
[after-name <name>]
```

#### Important:

You cannot assign both an after name and before name to the same IPsec policy.

#### Important:

You cannot assign an after name or before name to a previously existing IPsec policy.

To display a list of existing IPsec policies, use the **show crypto ipsec policy** command, with the **all** variable. For more information, see [Displaying IPsec policies](#) on page 247.

**Table 69: Variable definitions**

Variable	Value
after-name <name>	Specifies the existing IPsec policy after which to insert this new policy.
before-name <name>	Specifies the existing IPsec policy before which to insert this new policy.
<policy-name>	IPsec policy name. Max 8 characters.
<peer>	Specifies the peer IP address or the domain name for IKE negotiations.
If you create an IPsec policy without specifying a before name and an after name, the new policy is appended at the end of the list of existing IPsec policies.	

Variable	Value
	When you remove an IPsec policy, you do not have to specify an after name or before name.

## Configuring ABOT tunneling

The following procedure describes how to configure ABOT tunneling.

### Note:

When you create an ABOT tunnel on the SR2330/4134, by default the Dead Peer Detection mode is set to on demand. If you create a tunnel to an Avaya VPN router that has keepalives enabled, the Avaya VPN router will send keepalives every 60 seconds. When it does not get a response, it will tear down the tunnel with no teardown message.

In this case, you need to allow keepalives through the SR2330/4134 firewall, otherwise the SR2330/4134 will not respond to the Avaya VPN router keepalives and the tunnel will be torn down.

To allow the SR2330/4134 firewall to accept the Avaya VPN router keepalives, use the following command:

```
Host1/configure/firewall internet # policy 100 in service ike self
```

### Procedure Steps

1. To configure ABOT tunneling enhancements, enter Configuration Mode.

```
configure terminal
```

2. Enter the **crypto** subtree of commands.

```
crypto
```

3. Create a policy.

```
ike policy to-ces <address>
```

4. Configure the key-id to match.

```
local-id [key-id <key_id> | email-id <email> | domain-name  
<fqdn> | der-encoded-dn <name>]
```

5. Configure a local address.

```
local-address <local address>
```

6. Disable initial contact.

```
no initial-contact
```

7. Exit the **crypto** subtree.

```
exit
```

**Table 70: Variable definitions**

Variable	Value
<address>	Specifies the mapped address of the server.
der-encoded-dn <name>	Specifies the x.500 (LDAP) distinguished name.
domain-name <fqdn>	Specifies a fully qualified domain name (FQDN), like router.com.
email-id <email>	Specifies a fully-qualified email user name string, like name@router.com.
key-id <key_id>	Specifies the key to match. This is an optional parameter.
<local address>	<p>Specifies the local address of the server</p> <p>The <b>key-id</b> option for the <b>local-id</b> command should only be used when the local IP address under the IKE policy is configured as 0.0.0.0.</p> <p>The option for the command is available only when the local IP address is 0.0.0.0.</p> <p>When the local-address must be 0.0.0.0 under the IKE policy due to dynamic tunnels or the use of DHCP WAN address, you are prompted to enter the local ID before configuring the local IP address of 0.0.0.0.</p>
<p><b>Note:</b></p> <p>The Secure Router does allow one to enter a key-id with local-address being fixed-ip by changing the tunnel mode to aggressive and exchange-type to be initiator-only. If a user does this procedure, saves the config and reboots with that config, the Secure Router's config will not load the ike policy upon reboot and will throw an error message.</p> <p><b>Important:</b></p> <p>Avaya does not support using the key-id option for local-id with a fixed ip address under the IKE policy.</p>	

## Configuring anti-replay

Enable or disable anti-replay service on the inbound security association. The default is disabled.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-gateway-ip>
```

4. To enable or disable anti-replay, enter:

```
[no] anti-replay
```

---

## Enabling or disabling the IPsec policy entry

Enable or disable the IPsec policy entry.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-gateway-ip>
```

4. To enable or disable the IPsec policy, enter:

```
[no] enable
```

---

## Configuring an IPSec VPN bypass policy

Use this procedure to configure a policy that enables selected network traffic to bypass an IPSec VPN tunnel.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure as bypass, enter:

```
ipsec policy <policy-name> bypass
```

**Important:**

You cannot use `bypass` as a DNS name to refer to a peer in the `ipsec policy <policy name> <peer>` command.

4. To specify the IPsec bypass policy match criteria for network traffic, enter:

```
match address
<source-start-ip> <source-mask>
<dest-start-ip> <dest-mask>
[source-end-ip <A.B.C.D>]
[dest-end-ip <A.B.C.D>]
[protocol <protocol>]
[sport <0-65535>]
[dport <0-65535>]
```

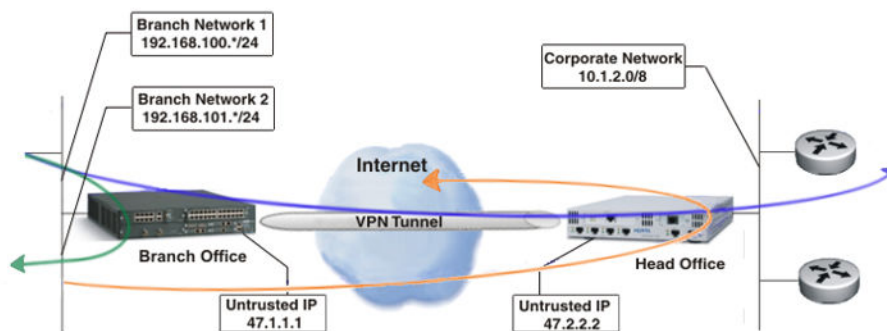
**Table 71: Variable definitions**

Variable	Value
<source-start-ip> <source-mask>	Source IP address and subnet mask of the IP stream that is to be protected by the IPsec policy. If you are defining a range of addresses, this represents the start address in the range.
<dest-start-ip> <dest-mask>	Destination IP address and subnet mask of the IP stream that is to be protected by the IPsec policy. If you are defining a range of addresses, this represents the start address in the range.
[source-end-ip <A.B.C.D>]	If you are defining a range of addresses for the source IP, this parameter specifies the end address in the range.
[dest-end-ip <A.B.C.D>]	If you are defining a range of addresses for the destination IP, this parameter specifies the end address in the range.
[protocol <protocol>]	Specifies a protocol for the IP stream to be protected. Values include: <ul style="list-style-type: none"> <li>• <code>udp</code>—UDP protocol</li> <li>• <code>tcp</code>—TCP protocol</li> <li>• <code>icmp</code>—ICMP protocol</li> <li>• <code>gre</code>—GRE protocol</li> <li>• <code>protocol-number</code>—the number of the protocol</li> </ul>
[sport <0-65535>]	Specifies a source port value for the IP stream to be protected.

Variable	Value
[dport <0-65535>]	Specifies a destination port value for the IP stream to be protected.

## Configuration example — IPsec VPN bypass policy

This section provides sample configuration for IPsec VPN bypass policies based on the network scenarios displayed in the following diagram.



### Configuration steps

1. To specify a bypass policy on traffic originating from Branch Network 1 (192.168.100.0/24) and destined to Branch Network 2 (192.168.101.0/24), enter:

```
ipsec policy local bypass
match address 192.168.100.0 24 192.168.101.0 24
```

2. To specify a bypass policy on traffic originating from Branch Network 2 (192.168.101.0/24) and destined to Branch Network 1 (192.168.100.0/24), enter:

```
ipsec policy local1 bypass
match address 192.168.101.0 24 192.168.100.0 24
```

3. To specify an IPsec policy to be applied to all traffic originating from the branch office and destined to the head office, enter:

```
ipsec policy toHead 47.2.2.2
match address 0.0.0.0 0 0.0.0.0 0
```

## Enabling IPsec nailed up tunnel

Use this procedure to configure an IPsec policy with the nailed up feature enabled.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To enable IPsec nailed up tunnel, enter:

```
nailed-up
```

**Table 72: Variable definitions**

Variable	Value
<policy-name>	Specifies the IPsec policy name. The maximum is 8 characters.
<peer-address>	Specifies the peer security gateway IP address.

### Job aid: IPsec nailed up tunnel sample configuration

The following job aid provides sample steps for configuring a branch office Secure Router to create connection that does not require data to establish a tunnel with a head office device.

1. To configure the IKE policy, enter:

```
configure terminal
crypto
ike policy test 20.1.1.2
local-address 20.1.1.1
key avaya123
proposal 1
exit proposal
exit policy
```

2. To configure the IPsec policy, enter:

```
ipsec policy test 20.1.1.2
match address 10.1.1.0 255.255.255.0 30.1.1.0 255.255.255.0
proposal 1 esp
exit proposal
nailed-up
exit policy
```

## Specifying the IP stream on which to apply IPsec

Specify the IP stream on which to apply IPsec.

When this command is entered, a default proposal is created, with the following properties: priority 1, ESP, 3DES, SHA1, DH-group2, tunnel mode.



## Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To specify crypto configuration for IPsec and IKE, enter:  

```
crypto
```
3. To specify the IPsec policy to configure, enter:  

```
ipsec policy <policy-name> <peer-gateway-ip>
```
4. To specify the IP stream on which to apply IPsec, enter:  

```
match address
<source-start-ip> <source-mask>
<dest-start-ip> <dest-mask>
[source-end-ip <A.B.C.D>]
[dest-end-ip <A.B.C.D>]
[protocol <protocol>]
[sport <0-65535>]
[dport <0-65535>]
```

**Table 73: Variable definitions**

Variable	Value
<source-start-ip> <source-mask>	Source IP address and subnet mask of the IP stream that is to be protected by the IPsec policy. If you are defining a range of addresses, this represents the start address in the range.
<dest-start-ip> <dest-mask>	Destination IP address and subnet mask of the IP stream that is to be protected by the IPsec policy. If you are defining a range of addresses, this represents the start address in the range.
[source-end-ip <A.B.C.D>]	If you are defining a range of addresses for the source IP, this parameter specifies the end address in the range.
[dest-end-ip <A.B.C.D>]	If you are defining a range of addresses for the destination IP, this parameter specifies the end address in the range.
[protocol <protocol>]	Specifies a protocol for the IP stream to be protected. Valid values are: <ul style="list-style-type: none"> <li>• udp—UDP protocol</li> <li>• tcp—TCP protocol</li> <li>• icmp—ICMP protocol</li> </ul>

Variable	Value
	<ul style="list-style-type: none"> <li>• gre—GRE protocol</li> <li>• protocol-number—the number of the protocol</li> </ul>
[sport <0-65535>]	Specifies a source port value for the IP stream to be protected.
[dport <0-65535>]	Specifies a destination port value for the IP stream to be protected.

## Configuring DH prime modulus group for PFS

Configure the Diffie-Hellman prime modulus group for Perfect Forward Secrecy (PFS).

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-gateway-ip>
```

4. To enable or disable anti-replay, enter:

```
pfs-group {group1 | group2 | group5 | group14 | group15}
```

**Table 74: Variable definitions**

Variable	Value
group1	768-bit. RFC 2409
group2	1024-bit. RFC 2409.
group5	1536-bit. RFC 2409.
group14	2048-bit modular exponential (MODP) group. RFC 2409.
group15	3072-bit MODP group. RFC 2409.

## Assigning multiple networks to a single IPsec policy

Use this procedure to assign a range of network addresses, specified as IP address objects, to an IPsec policy.

**Note:**

You can use `address <object-name> <ipaddress>` to configure multiple objects for a **range of IP addresses or subnets** depending on the subnet mask you enter. The router interprets `255.255.0.0` mask as a range and `16` as a subnet. For example, `address OFT 47.10.64.0 255.255.0.0` is treated as a range of addresses and `address OFT 47.10.64.0 16` as a subnet.

**Procedure steps**

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify global firewall configuration, enter:

```
firewall global
```

3. To specify firewall object configuration, enter:

```
object
```

4. To create an IP address object for a range of IP addresses, enter:

```
[no] [address <object-name> <ipaddress>]
```

5. To exit firewall configuration, enter:

```
exit firewall
```

6. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

7. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer>
```

8. To assign the IP address object to the IPsec policy, enter:

```
match object <source-object> <destination-object> [protocol  
<protocol>] [sport <0-65535>] [dport <0-65535>]
```

**Table 75: Variable definitions**

Variable	Value
[address <object-name> <ipaddress>]	<p>Specifies an IP address object. This value can be specified as a range of IP addresses [<code>&lt;start-address&gt; &lt;end-address&gt;</code>] or an IP address subnet [<code>address &lt;prefix-len&gt;</code>].</p> <p><b>Important:</b></p> <p>The Secure Router supports multiple networks in a single IPSec policy only when the IP address object is configured in the IP address range format.</p>

Variable	Value
<source-object>	Specifies the crypto object to match to the source IP address.
<destination-object>	Specifies the crypto object to match to the destination IP address.
[protocol <protocol>]	Specifies a protocol for the IP stream to be protected. Values include: <ul style="list-style-type: none"> <li>• udp—UDP protocol</li> <li>• tcp—TCP protocol</li> <li>• icmp—ICMP protocol</li> <li>• gre—GRE protocol</li> <li>• protocol-number—the number of the protocol</li> </ul>
[sport <0-65535>]	Specifies a source port. Values range from 0 to 65535.
[dport <0-65535>]	Specifies a destination port. Values range from 0 to 65535.

### Job aid: sample configuration

This section provides a sample configuration with a range of source and destination network IP addresses specified as IP address objects and those objects assigned to an IPsec policy.

1. To configure Ethernet interface 0/1, enter:

```
configure terminal
interface ethernet 0/1
ip address 10.1.1.10 255.255.255.0
crypto trusted
exit ethernet
```

2. To configure Ethernet interface 0/2, enter:

```
interface ethernet 0/2
ip address 100.1.1.1 255.255.255.0
crypto untrusted
exit ethernet
```

3. To configure Ethernet interface 0/3, enter:

```
interface ethernet 0/3
ip address 20.1.1.10 255.255.255.0
crypto trusted
exit ethernet
```

4. To configure the IP routes, enter:

```
ip route 50.1.1.0/24
ip route 60.1.1.0/24 100.1.1.2
```

5. To specify a range of source and destination network addresses as IP address objects in the global firewall, enter:

```
firewall global
object
address dest 50.1.1.0 50.1.1.255
address dest 60.1.1.0 60.1.1.255
address src 10.1.1.0 10.1.1.255
```

```
address src 20.1.1.0 20.1.1.255
exit object
```

6. To configure further global firewall parameters, enter:

```
algs
dns
exit dns
exit algs
max-connection-limit self 2048
exit firewall
```

7. To configure the internet firewall, enter:

```
firewall internet
interface ethernet0/2
policy 100 in permit self
exit policy
exit firewall
```

8. To configure the corporate firewall, enter:

```
firewall corp
interface ethernet0/1 ethernet0/3
policy 100 in permit
exit policy
policy 1024 out permit
exit policy
exit firewall
```

9. To configure the IKE policy, enter:

```
crypto
ike policy To-Head 100.1.1.2
local-address 100.1.1.1
key tasmanet
proposal 1
exit proposal
exit policy
```

10. To configure an IPSec policy and assign IP address objects to the policy, enter:

```
ipsec policy To-Head 100.1.1.2
match object src dest
exit policy
exit crypto
```

---

## Configuring IPsec proposal

Configure an IPsec proposal for an IPsec SA.

Before configuring a proposal, you must specify the IP stream on which to apply IPsec using the **match address** command.

In case multiple proposals are configured, all of them are sent in the SA payload in a logical OR manner in the order they are specified by the proposal priority. The protocol value defaults to ESP if it is not explicitly specified.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To select the IPsec proposal to configure, enter:

```
proposal <1-5> protocol {esp | ah}
```

**Table 76: Variable definitions**

Variable	Value
<1-5>	Specifies the proposal priority.
esp	Specifies ESP protocol. When chosen, the proposal parameters are set to the following defaults: 3DES, SHA1 and tunnel mode.
ah	Specifies AH protocol. When chosen, the proposal parameters are set to the following defaults: SHA1 and tunnel mode.

## Configuring encryption algorithm for IPsec proposal

Configure the encryption algorithm for the IPsec proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To select the IPsec proposal to configure, enter:

```
proposal <1-5>
```

5. To configure the encryption algorithm for the proposal, enter:

```
encryption-algorithm {des-cbc | 3des-cbc | aes128-cbc |  
aes192-cbc | aes256-cbc}
```

**Table 77: Variable definitions**

Variable	Value
des-cbc	Specifies DES-CBC encryption.
3des-cbc	Specifies 3DES-CBC encryption.
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.
null	Specifies no encryption.

## Configuring hash algorithm for IPsec proposal

Configure the hash algorithm for the IPsec proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IPsec policy to configure, enter:  
`ipsec policy <policy-name> <peer-address>`
4. To select the IPsec proposal to configure, enter:  
`proposal <1-5>`
5. To configure the hash algorithm, enter:  
`hash-algorithm {md5|sha1|null}`

**Table 78: Variable definitions**

Variable	Value
md5-hmac	A 128-bit message digest-RFC 1321 + RFC 2085
sha1-hmac	Secure Hash Standard: A 160-bit message digest-NIST,FIPS PUB 180-1
null	no authentication

## Configuring lifetime for IPsec proposal

Configure the lifetime of the IPsec SA. When the SA expires, it is replaced by a new negotiated SA or terminated.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IPsec policy to configure, enter:  
`ipsec policy <policy-name> <peer-address>`
4. To select the IPsec proposal to configure, enter:  
`proposal <1-5>`
5. To configure lifetime, enter:  
`lifetime {kilobytes <300-4194303> | seconds <300-864000>}`

**Table 79: Variable definitions**

Variable	Value
kilobytes <300-4194303>	Specifies the volume of traffic (in kilobytes) that can pass between IPsec peers using the given IPsec SA before that SA expires. Default: 4194303.
seconds <300-864000>	Specifies the number of seconds the IPsec SA can live before expiring. Default: 3600 seconds (1 hour).

## Configuring the IPsec encapsulation mode for the proposal

Configure the IPsec encapsulation mode for the proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the IPsec policy to configure, enter:  
`ipsec policy <policy-name> <peer-address>`



4. To select the IPsec proposal to configure, enter:

```
proposal <1-5>
```

5. To configure mode, enter:

```
mode {transport | tunnel}
```

**Table 80: Variable definitions**

Variable	Value
tunnel	Specifies tunnel mode. In tunnel mode the IP header of the packet is encapsulated into a new IP header with a routable destination IP address. Protection is offered for the complete packet. This is the default mode.
transport	Specifies transport mode. In transport mode, the old IP address is retained and the hash (in case of AH) is generated over the payload and delivered to the peer. The protection is offered only for the pay load.

## Configuring static weighted tunnels for tunnel failover

Use this procedure to specify primary and secondary tunnels to provide IPSec VPN tunnel failover for branch offices.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To apply primary and backup failover policies, enter:

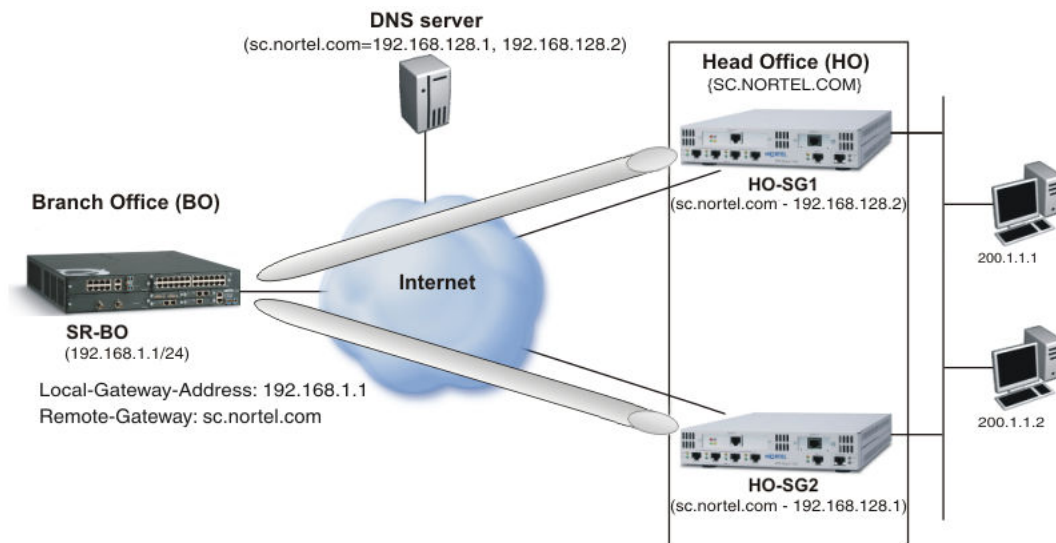
```
[no] failover <primary-policy-name> <backup-policy-name>
```

**Table 81: Variable definitions**

Variable	Value
primary-policy-name	Specifies the failover policy for the primary branch office tunnel. This policy must be configured as IPSec nailed up.
backup-policy-name	Specifies the failover policy for the secondary branch office tunnel.
<b>Important:</b> You cannot reuse any policies that were previously configured for failover.	

## Configuration example — Tunnel failover using round robin DNS

This configuration example provides sample configuration for tunnel failover using round robin DNS, based on the network topology shown in the following diagram.



**Figure 25: Tunnel failover using round robin DNS**

### Configuration scenario

The branch office Secure Router (SR-BO) is configured with a security policy database (SPD) policy for the head office (HO) network (192.168.128.0/24) and an associated IKE policy (local IP address – 192.168.1.1/24).

SR-BO is configured with the DNS name of the HO security gateway (sc.avaya.com).

HO has more than one tunnel endpoint to terminate the BO connections and the IP addresses of these endpoints are associated with the DNS name used by the BO.

The DNS server, which is external to SR-BO, is configured to service name resolution queries using a round robin algorithm, to make sure each query for a given DNS name is serviced with a different IP address than the previous query.

### Configuration steps

To configure the branch office Secure Router (SR-BO) for the scenario shown in the **Tunnel failover using round robin DNS** diagram above, perform the following steps.

1. To configure Ethernet interface 0/1, enter:

```
interface ethernet 0/1
ip address 192.168.1.1 255.255.255.0
```

```
crypto untrusted
exit Ethernet
```

2. To configure Ethernet interface 0/2, enter:

```
interface ethernet 0/2
ip address 10.1.1.1 255.255.255.0
crypto trusted
exit Ethernet
```

3. To configure the IP name server, enter:

```
ip pname_server 100.1.1.100
```

4. To configure the Internet firewall, enter:

```
firewall internet
interface ethernet0/1
policy 100 in permit self
exit policy
exit firewall
```

5. To configure the corporate firewall, enter:

```
firewall corp
interface ethernet0/2
policy 100 in permit
exit policy
policy 1024 out permit
exit policy
exit firewall
```

6. To configure the IKE policy, enter:

```
crypto
ike policy test sc.avaya.com
local-address 192.168.1.1
key tasmanet
mode aggressive
exchange-type initiator-only
proposal 1
exit proposal
exit policy
```

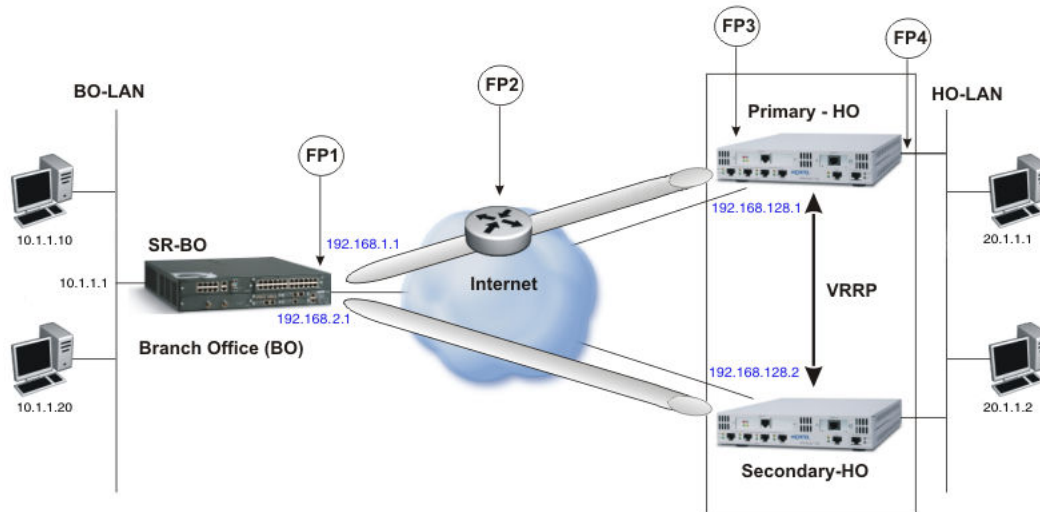
7. To configure the IPSec policy, enter:

```
ipsec policy toNVR sc.avaya.com
match address 10.1.1.0 255.255.255.0 200.1.1.0 255.255.255.0
proposal 1 esp
no esn
exit proposal
exit policy
```

---

## Configuration example — Tunnel failover using static weighted tunnels

This configuration example provides sample configuration for tunnel failover using static weighted tunnels, based on the network topology shown in the following diagram.



**Figure 26: Tunnel failover using static weighted tunnels**

### Network legend

BO – Branch Office  
 HO – Head Office  
 SR-BO – Branch Office device (Secure Router)  
 Primary-HO – Primary Head Office device (NVR)  
 Secondary-HO – Secondary Head Office device (NVR)  
 FP(n) – Failure points

### Configuration scenario

#### Failure Points

- FP1 – BO public interface
- FP2 – VPN tunnel path
- FP3 – HO public interface
- FP4 – HO private interface

#### SR-BO failure detection, failover, and recovery actions

- Detection — Dead Peer Detect (DPD) sends an Internet Key Exchange (IKE) informational message ("R\_U\_THERE") through the primary tunnel from SR-BO to Primary-HO. If Primary-HO fails to respond after several retries, SR-BO determines that a tunnel failure has occurred and flushes the IPsec, and IKE security association (SA) associated with the Primary tunnel.
- Failover — BO-LAN traffic destined for the HO-LAN matches the secondary security policy database (SPD) policy and triggers an IKE daemon to negotiate both an IKE and IPsec SA with Backup-HO.
- Recovery — The primary tunnel, configured as IPsec nailed up, retries the negotiation between SR-BO and Primary-HO every 20 seconds until IPsec SAs are setup. When SAs are established, BO-LAN traffic is diverted through the primary tunnel by flushing the IPsec and IKE SA of the secondary SPD policy.

## Primary-HO and Backup-HO failure detection, failover, and recovery actions

- Detection — Standards based DPD keepalive or local link failure detects the primary VPN tunnel failure.
- Failover — VRRP tracks the primary VPN tunnel status and switches over to Backup-HO.
- Recovery — The primary tunnel, configured as IPsec nailed up, retries persistently until connectivity is restored, and VRRP switches back to Primary-HO when the primary VPN tunnel status is UP.

## BO device configuration

- SR-BO is configured with two SPD policies. Both policies are destined for the same head office network (20.1.1.0/24) through two IPsec tunnels that secure traffic to the network.
- Each IPsec policy has a corresponding IKE policy.
- Each IPsec and IKE policy pair have a unique peer gateway in the head office through two IKE tunnels (192.168.1.1 -> 192.168.128.1 and 192.168.2.1 -> 192.168.128.2)
- The end user enables failover and identifies the primary and backup IPsec policies. Both policies are destined for the same head office network (20.1.1.0/24) through two IPsec tunnels that secure traffic to the network.
- The primary SPD policy must be configured as IPsec nailed up.

## Primary-HO device configuration

- The connection to BO is created in peer-to-peer mode, with the remote IP address as the BO public WAN interface IP address.
- IPsec nailed up is enabled for that connection group.
- VRRP is enabled on the private interface and serves as the VRRP master for the Virtual IP Address.
- The HO interface group is created by adding the BO IPsec tunnel to it.
- The HO interface group is registered for VRRP tracking.

## Backup-HO device configuration

- The connection to BO is created in peer-to-peer mode, with the remote IP address as the BO public interface IP address.
- VRRP is enabled on the private interface to backup the Virtual IP address for which the Primary-HO router is the master.

## Configuration steps

To configure the branch office Secure Router (SR-BO) for the scenario shown in the **Tunnel failover using static weighted tunnels** diagram above, perform the following steps.

1. To configure Ethernet interface 0/1, enter:

```
interface ethernet 0/1
ip address 192.168.2.1 255.255.255.0
```

```
crypto untrusted
exit Ethernet
```

2. To configure Ethernet interface 0/2, enter:

```
interface ethernet 0/2
ip address 10.1.1.1 255.255.255.0
crypto trusted
exit Ethernet
```

3. To configure the interface WAN bundle, enter:

```
interface bundle wan
link e1 3/1
encapsulation ppp
ip address 192.168.1.1 255.255.255.0
crypto untrusted
exit bundle
```

4. To configure the IP routes, enter:

```
ip route 20.1.1.0/24 wan
ip route 20.1.1.0/24 192.168.2.2 10
```

5. To configure the Internet firewall, enter:

```
firewall internet
interface ethernet0/1 wan
policy 100 in service ike self
exit firewall
```

6. To configure the corporate firewall, enter:

```
firewall corp
interface ethernet0/2
policy 100 in src 20.1.1.0 24 10.1.1.0 24

exit policy
exit firewall
```

7. To configure the primary IKE policy, enter:

```
crypto
ike policy primary 192.168.128.1
local-address 192.168.1.1
key avaya123
exit policy
```

8. To configure the backup IKE policy, enter:

```
ike policy backup 192.168.128.2
local-address 192.168.2.1
key avaya4567
exit policy
```

9. To configure the primary IPSec policy, enter:

```
ipsec policy primary 192.168.128.1
match address 10.1.1.0 255.255.255.0 20.1.1.0 255.255.255.0
nailed-up
exit policy
```

- To configure the backup IPSec policy, enter:

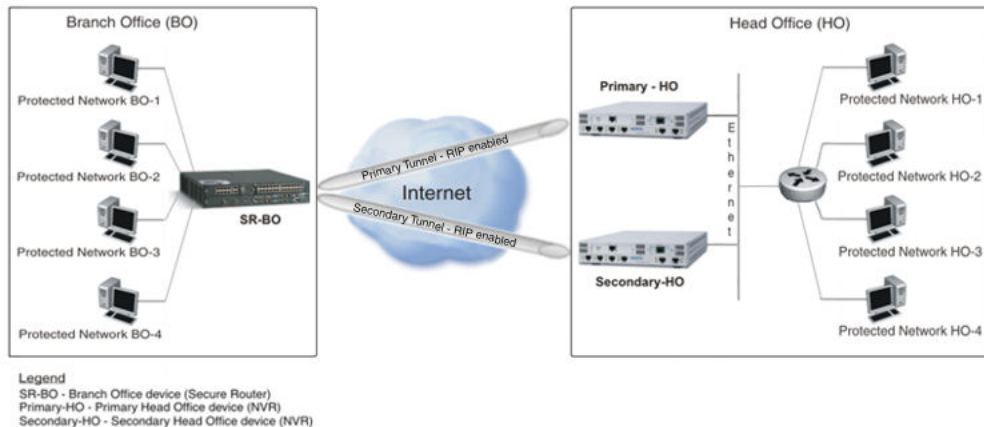
```
ipsec policy backup 192.168.128.2
match address 10.1.1.0 255.255.255.0 20.1.1.0 255.255.255.0
exit policy
```

- To configure tunnel failover, enter:

```
failover primary backup
```

## Configuration example 1

In this scenario, the branch office Secure Router, the primary head office NVR, and the secondary head office NVR are configured to use RIP.



- To configure Ethernet interface 0/2, enter:

```
interface ethernet 0/2
ip address 30.1.1.1 255.255.255.0
crypto trusted
exit ethernet
```

- To configure Ethernet interface 0/3, enter:

```
interface ethernet 0/3
ip address 8.8.8.1 255.255.255.0
crypto untrusted
exit ethernet
```

- To configure Ethernet interface 0/4, enter:

```
interface ethernet 0/4
ip address 50.1.1.1 255.255.255.0
crypto trusted
exit ethernet
```

- To configure the interface bundle, enter:

```
interface bundle priwan
link t1 2/2
encapsulation ppp
ip address 4.4.4.1 255.255.255.0
crypto untrusted
exit bundle
```

- To configure the primary tunnel, enter:

```
interface tunnel primary
ip address 112.1.1.1 255.255.255.0
```

```
tunnel source 4.4.4.1
tunnel destination 200.1.1.2
tunnel mode ipip
tunnel protection primary tasmanetbo2
crypto untrusted
exit tunnel
```

6. To configure the secondary tunnel, enter:

```
interface tunnel backup
ip address 113.1.1.1 255.255.255.0
tunnel source 8.8.8.1
tunnel destination 201.1.1.2
tunnel mode ipip
tunnel protection backup avaya123bo2
crypto untrusted
exit tunnel
```

7. To configure the access list, enter:

```
access-list acl permit any
```

8. To configure the IP routes, enter:

```
ip route 200.1.1.0/24 4.4.4.2
ip route 201.1.1.0/24 8.8.8.2
```

9. To configure RIP, enter:

```
router rip
network primary
network backup
offset-list acl in 7 backup
offset-list acl in 5 primary
redistribute connected
redistribute static
exit rip
```

10. To configure the Internet firewall, enter:

```
firewall internet
interface ethernet0/3 priwan primary backup
policy 100 in permit self
exit policy
exit firewall
```

11. To configure the head office firewall, enter:

```
firewall corp
interface ethernet0/2 ethernet0/4
policy 100 in permit
exit policy
policy 1024 out permit
exit policy
exit firewall
```

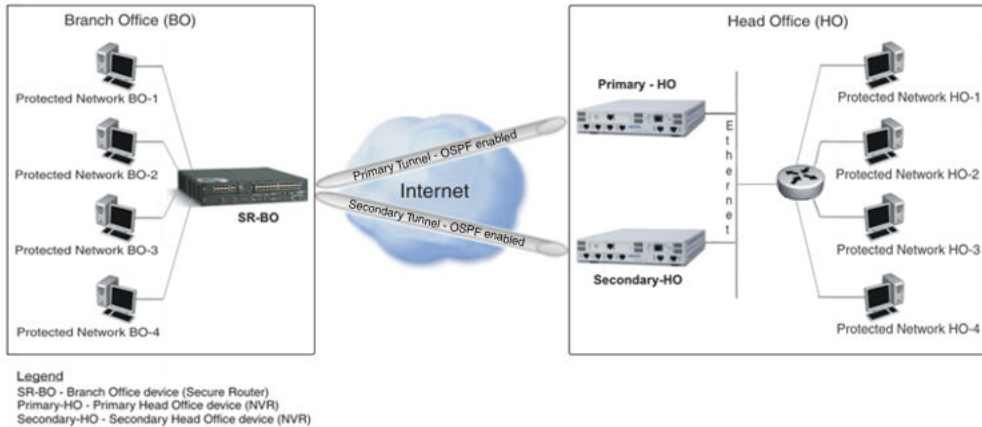
12. To configure keepalives, enter:

```
no keepalive mode on-demand
keepalive transmit-interval 10
```

## Configuration example 2

In this scenario, the branch office Secure Router, the primary head office NVR, and the secondary head office NVR are configured to use OSPF.





1. To configure Ethernet interface 0/2, enter:

```
interface ethernet 0/2
ip address 30.1.1.1 255.255.255.0
crypto trusted
exit ethernet
```

2. To configure Ethernet interface 0/3, enter:

```
interface ethernet 0/3
ip address 8.8.8.1 255.255.255.0
crypto untrusted
exit ethernet
```

3. To configure Ethernet interface 0/4, enter:

```
interface ethernet 0/4
ip address 50.1.1.1 255.255.255.0
crypto trusted
exit ethernet
```

4. To configure the interface bundle, enter:

```
interface bundle priwan
link t1 2/2
encapsulation ppp
ip address 4.4.4.1 255.255.255.0
crypto untrusted
exit bundle
```

5. To configure the primary tunnel, enter:

```
interface tunnel primary
ip address 112.1.1.1 255.255.255.0
ip ospf mtu-ignore
ip ospf cost 10
tunnel source 4.4.4.1
tunnel destination 200.1.1.2
tunnel mode ipip
tunnel protection primary tasmanntbo2
crypto untrusted
exit tunnel
```

6. To configure the secondary tunnel, enter:

```
interface tunnel backup
ip address 113.1.1.1 255.255.255.0
ip ospf mtu-ignore
ip ospf cost 20
```

```
tunnel source 8.8.8.1
tunnel destination 201.1.1.2
tunnel mode ipip
tunnel protection backup avaya123bo2
crypto untrusted
exit tunnel
```

7. To configure the router ID, enter:

```
router-id 4.4.4.1
```

8. To configure the IP routes, enter:

```
ip route 200.1.1.0/24 4.4.4.2
ip route 201.1.1.0/24 8.8.8.2
```

9. To configure OSPF, enter:

```
router ospf 1
redistribute connected
redistribute static
log-adjacency-changes
network 112.1.1.0 0.0.0.255 area 0
network 113.1.1.0 0.0.0.255 area 0
exit ospf
```

10. To configure the head office firewall, enter:

```
firewall corp
interface ethernet0/2 ethernet0/4
policy 100 in permit
exit policy
policy 1024 out permit
exit policy
exit firewall
```

11. To configure keepalives, enter:

```
no keepalive mode on-demand
keepalive transmit-interval 10
```

---

## Configuring remote access IKE policies

---

### Creating an IKE policy for remote access VPN

Create a dynamic IKE policy for remote access VPN.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of dynamic IKE policies for remote access, enter:

```
dynamic
```

4. To create a remote access IKE policy, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

**Table 82: Variable definitions**

Variable	Value
<policy-name>	Specifies the IKE policy name.
group-type {modecfg-group   l2tp-group}	modecfg-group: Mode config group. To configure the mode config client parameters, use the <b>ike policy client configuration</b> and <b>ike policy client authentication</b> commands. l2tp-group: L2TP group. To configure the L2TP server and client parameters, use the <b>l2tp-server</b> commands.

## Configuring the IKE policy local address for remote access VPN

Configure the local address to be used in IKE negotiations.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To configure the local address, enter:

```
local-address <A.B.C.D>
```

## Configuring the IKE policy local ID for remote access VPN

Configure the local ID to specify the IPsec identifiers for the host that is used in the identification payload during IKE negotiation.

**Procedure steps**

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To configure the local ID, enter:

```
local-id {domain-name <fqdn> | email-id <email> | der-encoded-dn <name>}
```

**Table 83: Variable definitions**

Variable	Value
domain-name <fqdn>	Specifies a fully qualified domain name (FQDN), like router.com.
email-id <email>	Specifies a fully-qualified email user name string, like name@router.com.
der-encoded-dn <name>	Specifies the x.500 (LDAP) distinguished name.

**Configuring the IKE policy remote ID for remote access VPN**

Configure the remote ID to specify the IPsec peer that participates in the IKE negotiation.

**Procedure steps**

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To configure the remote ID, enter:

```
remote-id {domain-name "<fqdn>" | email-id "<email>" | der-
encoded-dn "<name>"}
```

**Table 84: Variable definitions**

Variable	Value
domain-name "<fqdn>"	Specifies the fully qualified domain name (FQDN), like router.com. The value must be specified within quotes.
email-id "<email>"	Specifies a fully-qualified email user name string, like name@router.com. The value must be specified within quotes.
der-encoded-dn "<name>"	Specifies the x.500 (LDAP) distinguished name. The value must be specified within quotes.

## Configuring IKE mode for remote access VPN

Configure the IKE mode for the remote access policy.

### Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To specify crypto configuration for IPsec and IKE, enter:  

```
crypto
```
3. To specify configuration of remote access IKE policies, enter:  

```
dynamic
```
4. To specify the remote access IKE policy to configure, enter:  

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```
5. To configure the IKE mode, enter:  

```
mode {main | aggressive}
```

**Table 85: Variable definitions**

Variable	Value
main	Specifies use of full negotiation to establish a security association. Main mode provides identity protection.
aggressive	Specifies use of quick negotiation to establish a security association. Aggressive mode does not provide identity protection. (This is the default mode.)

## Configuring mode-config client parameters for remote access VPN

Configure the parameters for mode-config.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of remote access IKE policies, enter:  
`dynamic`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name> modecfg-group`
5. To specify client configuration, enter:  
`client configuration`
6. To configure the mode-config address pool, enter:  
`address-pool <1-3> <start-ip> <end-ip>`

**Table 86: Variable definitions**

Variable	Value
<1-3>	pool number
<start-ip>	start IP address
<end-ip>	end IP address

### Configuring DNS server address for mode config

Configure the DNS server address for mode config.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of remote access IKE policies, enter:  
`dynamic`
4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> modecfg-group
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure the mode-config parameters:

```
dns-server <primary-server-ip> <secondary-server-ip>
```

### Configuring WINS server address for mode config

Configure the WINS server address for mode config.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> modecfg-group
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure the mode-config parameters:

```
wins-server <primary-server-ip> <secondary-server-ip>
```

### Configuring client authentication for mode config

Configure the authentication method for the remote client.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> modecfg-group
```

5. To specify client configuration, enter:

```
client authentication
```

- To configure the client authentication, enter:

```
radius {pap | chap}
```

**Table 87: Variable definitions**

Variable	Value
pap	RADIUS-PAP authentication method.
chap	RADIUS-CHAP authentication method.

## Configuring the IKE pre-shared key for remote access VPN

Define a preshared key for the remote access IKE policy. The key is valid only when the proposal configured has the authentication method as pre-shared-key.

### Procedure steps

- To enter the configuration mode, enter:

```
configure terminal
```

- To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

- To specify configuration of remote access IKE policies, enter:

```
dynamic
```

- To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

- To configure the pre-shared key, enter:

```
key <key-string>
```

**Table 88: Variable definitions**

Variable	Value
<key-string>	key string, max 49 characters.

## Enabling OCSP on the remote access IKE policy

Enable OCSP to instruct the router to contact the CA for verification of the status of any certificate that the router receives.



**Procedure steps**

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of remote access IKE policies, enter:  
`dynamic`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name> {modecfg-group | l2tp-group}`
5. To enable or disable OCSP, enter:  
`[no] ocsp`

**Configuring PFS for remote access IKE policy**

Enable or disable Perfect Forward Secrecy (PFS) of both keys and identities (RFC 2409) for the remote access IKE policy.

**Procedure steps**

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of remote access IKE policies, enter:  
`dynamic`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name> {modecfg-group | l2tp-group}`
5. To enable or disable PFS:  
`[no] pfs`

---

**Configuring an IKE proposal for remote access VPN**

Configure an IKE proposal for remote access VPN.

**Procedure steps**

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To create the IKE proposal, enter:

```
proposal <1-5>
```

**Table 89: Variable definitions**

Variable	Value
<1-5>	Proposal value. Only one proposal is allows in aggressive mode.

## Configuring authentication method for IKE proposal for remote access VPN

Configure the IKE proposal authentication method for remote access VPN.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To specify the IKE proposal, enter:

```
proposal <1-5>
```

6. To configure the remote ID, enter:

```
authentication-method {pre-shared-key | dss-signature | rsa-signature}
```

**Table 90: Variable definitions**

Variable	Value
pre-shared-key	Specifies authentication using a pre-shared key, derived out of band
dss-signature	Specifies authentication using Digital Signature Standard.
rsa-signature	Specifies authentication using RSA Signature.

## Configuring DH group for IKE proposal for remote access VPN

Configure the DH group for the IKE proposal

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of remote access IKE policies, enter:  
`dynamic`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name> {modecfg-group | l2tp-group}`
5. To specify the IKE proposal, enter:  
`proposal <1-5>`
6. To configure DH group for IKE proposal  
`dh-group {group1 | group2 | group5}`

**Table 91: Variable definitions**

Variable	Value
group1	768-bit. RFC 2409.
group2	1024-bit. RFC 2409.
group5	1536-bit. RFC2409. This is the highest level of security and requires more processing time than group 1 and group 2.

## Configuring encryption algorithm for IKE proposal for remote access VPN

Configure the encryption algorithm for the remote access IKE proposal.

## Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

5. To specify the IKE proposal, enter:

```
proposal <1-5>
```

6. To configure the encryption algorithm for IKE:

```
encryption-algorithm {des-cbc | 3des-cbc | aes128-cbc |  
aes192-cbc | aes256-cbc}
```

**Table 92: Variable definitions**

Variable	Value
des-cbc	Specifies DES-CBC encryption.
3des-cbc	Specifies 3DES-CBC encryption.
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.

## Configuring IKE hash algorithm for remote access VPN

Configure the IKE authentication algorithm for a given remote access proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of remote access IKE policies, enter:

```
dynamic
```

- To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

- To specify the IKE proposal, enter:

```
proposal <1-5>
```

- To configure the hash algorithm for IKE:

```
hash-algorithm {md5|sha1}
```

**Table 93: Variable definitions**

Variable	Value
md5	A 128-bit message digest-RFC 1321
sha1	Secure Hash Standard: A 160-bit message digest-NIST,FIPS PUB 180-1

## Configuring IKE lifetime for remote access VPN

Configure the lifetime of the remote access IKE SA. When the SA expires, it is replaced by a new negotiated SA or terminated.

### Procedure steps

- To enter the configuration mode, enter:

```
configure terminal
```

- To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

- To specify configuration of remote access IKE policies, enter:

```
dynamic
```

- To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> {modecfg-group | l2tp-group}
```

- To specify the IKE proposal, enter:

```
proposal <1-5>
```

- To configure the lifetime, enter:

```
lifetime {kilobytes <300-4194303> | seconds <300-864000>}
```

**Table 94: Variable definitions**

Variable	Value
kilobytes <300-4194303>	Lifetime in kilobytes. Default: unlimited.

Variable	Value
seconds <300-864000>	Lifetime in seconds. Default: 86400 seconds.

---

## Configuring remote access IPsec policies

---

### Creating an IPsec policy for remote access VPN

Create an IPsec policy for a remote access IPsec SA.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To create a remote access IPsec policy, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`

**Table 95: Variable definitions**

Variable	Value
group-type [modecfg-group   l2tp-group]	modecfg-group: Mode config group. To configure the mode config client parameters, use the <b>ike policy client</b> commands. l2tp-group: L2TP group. To configure the L2TP server and client parameters, use the <b>l2tp-server</b> commands.

---

### Specifying the IP stream on which to apply IPsec for remote access VPN

Specify the IP stream on which to apply IPsec

When this command is entered, a default proposal is created, with the following properties: priority 1, ESP, 3DES, SHA1, DH-group2, tunnel mode.

## Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To specify the remote access IPsec policy to configure, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`
5. To configure the matching IP address, enter:  
`match {address <A.B.C.D> <mask>}`  
`[source-end-ip <A.B.C.D>]`  
`[dest-start-ip <A.B.C.D>]`  
`[dest-netmask <A.B.C.D>]`  
`[dest-end-ip <A.B.C.D>]`  
`[protocol <protocol>]`  
`[sport <0-65535>]`  
`[dport <0-65535>]`

**Table 96: Variable definitions**

Variable	Value
[address <A.B.C.D> <mask>]	source IP address (start address if range is applicable) in the IP stream to be applied IPsec.
[source-end-ip <A.B.C.D>]	source IP address (end address if range is applicable) in the IP stream to be applied IPsec
[dest-start-ip <A.B.C.D>]	destination IP address (start address if range is applicable) in the IP stream to be applied IPsec
[dest-netmask <A.B.C.D>]	Subnet mask.
[dest-end-ip <A.B.C.D>]	destination IP address (end address if range is applicable) in the IP stream to be applied IPsec
[protocol <protocol>]	udp udp protocol tcp tcp protocol icmp icmp protocol any all the protocols
[sport <0-65535>]	Source port value.
[dport <0-65535>]	Destination port value.

---

## Configuring DH prime modulus group for PFS

Configure the Diffie-Hellman prime modulus group for Perfect Forward Secrecy (PFS). This specifies the strength of the PFS group which the IPsec (phase 2) policy uses.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To specify the remote access IPsec policy to configure, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`
5. To configure the PFS group, enter:  
`pfs-group {group1 | group2 | group5}`

**Table 97: Variable definitions**

Variable	Value
group1	768-bit. RFC 2409
group2	1024-bit. RFC 2409.
group5	1536-bit. RFC 2409. This is the highest level of security and requires more processing time than group 1 and group 2.

---

## Configuring IPsec proposal template for remote access VPN

Configure IPsec proposal template

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:



```
dynamic
```

4. To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name> {modecfg-group | l2tp-group}
```

5. To configure IPsec proposal template, enter:

```
proposal <1-5> protocol {esp | ah}
```

**Table 98: Variable definitions**

Variable	Value
1-5	proposal priority
protocol [esp   ah]	esp: ESP ah: AH

## Configuring encryption algorithm for IPsec proposal for remote access VPN

Configure the encryption algorithm for the remote access IPsec proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify configuration of dynamic IKE policies for remote access, enter:

```
dynamic
```

4. To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name> {modecfg-group | l2tp-group}
```

5. To specify the IPsec proposal template to configure, enter:

```
proposal <1-5>
```

6. To configure the encryption algorithm, enter:

```
encryption-algorithm {des-cbc | 3des-cbc | aes128-cbc |  
aes192-cbc | aes256-cbc | null}
```

**Table 99: Variable definitions**

Variable	Value
des-cbc	Specifies DES-CBC encryption.
3des-cbc	Specifies 3DES-CBC encryption.
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.

Variable	Value
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.
null	Specifies no encryption.

## Configuring hash algorithm for IPsec proposal for remote access VPN

Configure the hash algorithm for the remote access IPsec proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To specify the remote access IPsec policy to configure, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`
5. To specify the IPsec proposal template to configure, enter:  
`proposal <1-5>`
6. To configure the hash algorithm, enter:  
`hash-algorithm {md5-hmac | sha1-hmac | null}`

**Table 100: Variable definitions**

Variable	Value
md5	A 128-bit message digest-RFC 1321
sha1	Secure Hash Standard: A 160-bit message digest-NIST,FIPS PUB 180-1
null	No authentication

## Configuring lifetime for IPsec proposal for remote access VPN

Configure the lifetime of the remote access IPsec SA. When the SA expires, it is replaced by a new negotiated SA or terminated.

## Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To specify the remote access IPsec policy to configure, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`
5. To specify the IPsec proposal template to configure, enter:  
`proposal <1-5>`
6. To configure lifetime:  
`lifetime {kilobytes <300-4194303> | seconds <300-864000>}`

**Table 101: Variable definitions**

Variable	Value
kilobytes <300-4194303>	Lifetime in kilobytes. Default: 4194303.
seconds <300-864000>	Lifetime in seconds. Default: 3600 seconds.

## Configuring the IPsec encapsulation mode for a remote access proposal

Configure the IPsec encapsulation mode for the remote access proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify configuration of dynamic IKE policies for remote access, enter:  
`dynamic`
4. To specify the remote access IPsec policy to configure, enter:  
`ipsec policy <name> {modecfg-group | l2tp-group}`
5. To specify the IPsec proposal template to configure, enter:

```
proposal <1-5>
```

- To configure the encapsulation mode, enter:

```
mode {transport | tunnel}
```

**Table 102: Variable definitions**

Variable	Value
tunnel	Specifies tunnel mode. In tunnel mode the IP header of the packet is encapsulated into a new IP header with a routable destination IP address. Protection is offered for the complete packet. This is the default mode.
transport	Specifies transport mode. In transport mode, the old IP address is retained and the hash (in case of AH) is generated over the payload and delivered to the peer. The protection is offered only for the pay load.

## Enabling the dynamic IPsec policy

Enable the dynamic IPsec policy.

### Procedure steps

- To enter the configuration mode, enter:

```
configure terminal
```

- To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

- To specify configuration of dynamic IKE policies for remote access, enter:

```
dynamic
```

- To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name> {modecfg-group | l2tp-group}
```

- To enable the policy, enter:

```
enable
```

**Table 103: Variable definitions**

Variable	Value
[no]	Disables the policy.

---

## Configuring L2TP server for L2TP remote access

---

---

### Creating the L2TP remote access interface

Configure the L2TP server.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To create the L2TP access virtual interface, enter:  
`interface l2tp-server <server-name>`

---

### Configuring IP address for the L2TP access interface

Configure the IP address for the L2TP server.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To select the L2TP access virtual interface, enter:  
`interface l2tp-server <server-name>`
3. To configure the IP address of the L2TP server, enter:  
`ip address <ipaddress>`

---

### Configuring IPsec protection for the L2TP access interface

Configure IPsec protection for the L2TP server.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To select the L2TP access virtual interface, enter:

```
interface l2tp-server <server-name>
```

3. To configure IPsec protection, enter:

```
ipsec-protection <ike-ipsec-policy>

<untrusted-if-address>

{remote-id-type {ip-address | domain-name | email-id|der-
encoded-dn}}

[remote-id-data <remote-id-data>]

[key <key>]
```

**Table 104: Variable definitions**

Variable	Value
<ike-ipsec-policy>	Name of crypto dynamic IKE and IPsec policy. Max 8 characters.
<untrusted-if-address>	Address of the local crypto untrusted interface that is used as the IKE authenticated tunnel endpoint.
{remote-id-type {ip-address   domain-name   email-id  der-encoded-dn}}	Remote ID type: ip-address: a routeable IP address already configured on this device domain-name: fully qualified domain name (FQDN) email-id: email address (user FQDN) der-encoded-dn: x509 certificate subject-name in ascii form (default) (example: O=ACME Corp,OU=*,C=US,CN=Wile E. Coyote)
[remote-id-data <remote-id-data>]	Remote ID data. Max 48 characters.
[key <key>]	Tells IKE to use preshared-key authentication with this key (no key means use certificates).

## Configuring client parameters for L2TP remote access

Configure the client parameters for remote access with L2TP.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To select the L2TP access virtual interface, enter:

```
interface l2tp-server <server-name>
```

3. To select L2TP client configuration, enter:

```
remote-config {domain-name <domain-name>}
```

4. To configure L2TP client parameters, enter:

```
{address-pool <first-address> <last-address> | dns <A.B.C.D>
| nbns <A.B.C.D>}
```

**Table 105: Variable definitions**

Variable	Value
address-pool <first-address> <last-address>	Pool of IP addresses given to connecting clients.
dns <A.B.C.D>	DNS address the client will use.
nbns <A.B.C.D>	Net Bios Name Server (WINS server) the client will use.

## Configuring user parameters for L2TP remote access

Configure user parameters for remote access using L2TP.

### Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To select the L2TP access virtual interface, enter:  

```
interface l2tp-server <server-name>
```
3. To configure L2TP user parameters, enter:  

```
remote-user <username> <password>
```

## Shutting down the L2TP access interface

Shut down the L2TP server.

### Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To select the L2TP access virtual interface, enter:  

```
interface l2tp-server <server-name>
```
3. To shut down the interface, enter:  

```
shutdown
```

---

## Configuring dead peer detection keepalive

---

### Enabling dead peer detection

Use this procedure to enable or disable dead peer detection (DPD).

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To enable or disable dead peer detection keepalive, enter:  
`[no] keepalive enable`

---

### Configuring the DPD mode

Use this procedure to configure the Secure Router to use on-demand or periodic DPD for querying the operational status of IKE peers.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To enable periodic keep alive mode, enter:  
`keepalive mode <on-demand> <periodic>`

**Table 106: Variable definitions**

Variable	Value
on-demand	When selected (default), DPD retries are sent on demand.
periodic	When selected DPD retries are sent at regular intervals.



---

## Configuring the DPD keepalive retry interval

Use this procedure to configure the time interval the Secure Router waits between DPD keepalive retry messages.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To configure the dead peer detection keep alive retry interval, enter:

```
keepalive retry-interval <retry-interval>
```

**Table 107: Variable definitions**

Variable	Value
<retry-interval>	Specifies the number of seconds (configured in multiples of 10) between DPD keepalive retries. The default value is 10 seconds.

---

## Configuring the DPD keepalive transmit interval

Use this procedure to configure the time period the Secure Router waits for a response from an IKE peer before transmitting a DPD keepalive message.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To configure the disable dead peer detection keepalive transmit interval, enter:

```
keepalive transmit-interval <10-3600>
```

**Table 108: Variable definitions**

Variable	Value
< 10-3600>	<p>Specifies the DPD keepalive transmit interval (configured in multiples of 10 seconds).</p> <ul style="list-style-type: none"> <li>• On-demand DPD – this value determines the amount of time the Secure Router waits for a response from an IKE peer before sending a keepalive message to the peer.</li> <li>• Periodic DPD - this value determines the time interval for the Secure Router to send regular operational status queries to IKE peers. Values range from 10 to 3600 seconds. The default value is 10 seconds.</li> </ul>

---

## Configuring PMTU

---

### Configuring DF bit

Configure the value of the Don't Fragment (DF) bit for an interface.

The DF bit value for an interface can be set, copy, or clear. If the DF bit value is configured as **set** for an interface, it forces PMTU discovery regardless of the value of the DF bit value in the clear packet. If the DF bit value is configured as **copy**, this enables PMTU only when the clear packet is generated with the DF bit set. If the DF bit value is configured as **clear**, PMTU is not enabled.

#### Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To specify crypto configuration for IPsec and IKE, enter:  

```
crypto
```
3. To configure df bit, enter:  

```
pmtu df-bit {set | copy | clear} <if-name>
```

---

## Configuring the MTU threshold value

Configure the MTU threshold value. If the MTU value received in “ICMP PMTU” is less than the threshold, PMTU discovery is reset for that particular SA. That is, the DF bit is cleared for the particular SA.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To configure threshold MTU value, enter:  
`pmtu threshold-mtu <mtu-value>`

---

## Configuring processing of unsecured ICMP messages

Enable or disable processing of clear, unsecured ICMP messages.

Disable processing of unsecured ICMP messages to allow processing of only secured packets.

This option is disabled by default.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To enable or disable processing of unsecured ICMP messages, enter:  
`[no] pmtu unsecured-icmp-processing`

---

## Configuring TCP MSS on an Ethernet interface

Use this procedure to configure the TCP MSS value on an Ethernet interface.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To select the ethernet, enter:  
`interface ethernet <slot/port number>`
3. To specify the tcp-mtu of the ethernet, enter:  
`ip tcp-mss <value>`
4. To exit the ethernet configuration mode, enter:  
`exit`

**Table 109: Variable definitions**

Variable	Value
ip tcp-mss <value>	Specifies the TCP MSS clamping value for the tunnel. Values range from 536 to 9176.
<slot/port number>	Specifies the chassis slot number and port number for the Ethernet interface.

---

## Configuring CA trustpoint

Configure parameters for a specific certificate authority.

---

## Configuring the certificate enrollment method

Configure the certificate enrollment method.

To modify a pre-configured enrollment method for a trustpoint, you must delete the trustpoint and reconfigure it for the new enrollment method.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint to configure, enter:

```
ca trustpoint <ca-name>
```

4. To select the style of enrollment as either the manual cut and paste method or the automatic SCEP method, enter:

```
enrollment {terminal | url <url> }
```

**Table 110: Variable definitions**

Variable	Value
url <url>	Specifies the SCEP server URL through which the Certificate Authority can be accessed for submitting certificate requests or retrieving the approved certificates through SCEP. Specify the complete URL (with http://), including the cgi path and server port number, if any.
terminal	Specifies cut and paste mode for enrollment.

## Configuring parameters for the certificate request

### Configuring the certificate subject name

Specify the subject name that identifies the Secure Router in the certificate request.

#### Procedure steps

1. To enter the configuration mode, enter:  

```
configure terminal
```
2. To specify crypto configuration for IPsec and IKE, enter:  

```
crypto
```
3. To specify the CA trustpoint to configure, enter:  

```
ca trustpoint <ca-name>
```
4. To configure the subject name, enter:  

```
subject-name "<subject-name>"
```

### Configuring IP address as subjectAltName for the certificate

Specify the IP address as the subjectAltName to be used in the certificate request. To clear the IP address from the configuration, use the no form of the command.

In addition to the subject name, the certificate can support a SubjectAltName as an alternate means of identifying the router.

**Procedure steps**

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint to configure, enter:  
`ca trustpoint <ca-name>`
4. To configure the IP address, enter:  
`[no] ip-address <A.B.C.D>`

**Configuring fully-qualified domain name as subjectAltName for the certificate**

Specify the fully qualified domain name as the subjectAltName to be used in the certificate request. To clear the FQDN from the configuration, use the no form of the command.

**Procedure steps**

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint to configure, enter:  
`ca trustpoint <ca-name>`
4. To configure the fully-qualified domain name, enter:  
`[no] fqdn <fqdn>`

**Configuring e-mail address as subjectAltName for the certificate**

Specify the email address (user fully qualified domain name) as the subjectAltName that is used in the certificate request. To clear the email address from the configuration, use the no form of the command.

**Procedure steps**

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`

3. To specify the CA trustpoint to configure, enter:

```
ca trustpoint <ca-name>
```

4. To configure the e-mail address, enter:

```
[no] email <email-address>
```

## Configuring key pair for the certificate

Specify the key pair details like key ID, signature algorithm (RSA/DSA) and key size to be used in the certificate request. To clear the key pair details from the configuration, use the no form of the command.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the CA trustpoint to configure, enter:

```
ca trustpoint <ca-name>
```

4. To configure the key pair, enter:

```
[no] keypair <keypair-name> {rsa | dsa} {512 | 1024}
```

**Table 111: Variable definitions**

Variable	Value
<keypair-name>	Specifies the name of the key pair.
{rsa   dsa}	Specifies the key type. Values include: <ul style="list-style-type: none"> <li>• rsa – the RSA public key encryption algorithm</li> <li>• dsa – Digital Signature Algorithm</li> </ul>
{512   1024   2048   3072   4096}	Specifies the maximum supported key sizes. Values include: <ul style="list-style-type: none"> <li>• rsa - 512, 1024, 2048, 3072, and 4096 bits. The default is 1024 bits.</li> <li>• dsa - 512, 1024 bits. The default is 1024 bits.</li> </ul>

---

## Configuring certificate password

Although it is not a common scenario, depending upon the policy of the CA, the certificates issued by the CA can be encrypted. In cases where the certificate is encrypted, then a password is needed to decrypt them. This password is defined here.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint to configure, enter:  
`ca trustpoint <ca-name>`
4. To configure the password, enter:  
`password <password>`

---

## Authenticating the CA and importing a CA certificate

Authenticate the certificate authority and import the CA certificate. Importing CA certificate varies depending on the configured enrollment method (terminal or SCEP).

If the enrollment method is SCEP, the system imports the digital certificate of the CA using the SCEP protocol. And if the enrollment method is terminal, the system prompts you to paste the certificate obtained from the CA.

In order to authenticate the CA, you should verify the fingerprint of the imported certificate with the actual fingerprint of the CA obtained through any out-of-band mechanism.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint, enter:  
`ca authenticate <ca-name>`



If SCEP enrollment, the system imports the digital certificate of the CA using SCEP. If manual enrollment is specified, the system prompts you to enter the certificate.

4. To save the certificate to flash memory, enter:

```
save local
```

---

## Generating a certificate request for enrollment

Generate a certificate request for enrollment with the certificate authority. This command interactively collects the certificate information from the user, generates the private and public keys, and generates the certificate request in the required format. The generated certificate request is submitted to the CA through the configured enrollment method.

If the configured enrollment method is SCEP, then, the certificate request is automatically submitted to the CA. If the enrollment method is manual, the certificate request is printed on the standard output in the required format.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To generate the certificate request, enter:

```
ca enroll <ca-name>
```

---

## Manually importing a self certificate

If you are not using SCEP to import certificates, you can manually import the router certificate into the router using the cut and paste method.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the CA trustpoint from which the certificate was issued, enter:

```
ca import <ca-name>
```

4. To manually import the self certificate, enter:

```
router-certificate
```

The system prompts you to enter the certificate.

---

## Automatically importing a self certificate through SCEP

The following procedure provides a summary of the steps required to import a self certificate using SCEP.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the CA trustpoint, enter:

```
ca authenticate <ca-name>
```

4. To specify the SCEP server URL, enter:

```
enrollment url <url> exit
```

5. To authenticate the CA trustpoint, enter:

```
ca authenticate <ca-name>
```

6. To generate the certificate request, enter:

```
ca enroll <ca-name>
```

7. To save the configuration, enter:

```
save local
```

**Table 112: Variable definitions**

Variable	Value
<ca-name>	Specifies the name of the Certificate Authority.
url <url>	Specifies the SCEP server URL through which the Certificate Authority can be accessed for submitting certificate requests or retrieving the approved certificates through SCEP. Specify the complete URL

Variable	Value
	(with http://), including the cgi path and server port number, if any.

---

## Manually importing an OCSP Responder certificate

Manually import the OCSP Responder certificate into the router using the cut and paste method. This procedure is optional and is used only if the OCSP Responder does not send the responder certificate along with the certificate status information. To delete the responder certificate, use the `clear crypto ca certificates` command.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint from which the certificate was issued, enter:  
`ca import <ca-name>`
4. To manually import the responder certificate, enter:  
`responder-certificate`  
The system prompts you to enter the certificate.

---

## Configuring LDAP parameters

Configure the LDAP server URL to retrieve the CRL (certificate revocation list) from the directory database server. The LDAP client supports periodic downloads of the CRL using the “next update” time as specified by the CA in the CRL.

If the LDAP server URL is not configured, the CRL is retrieved through SCEP or using the manual (cut and paste) method.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the CA trustpoint to configure, enter:

```
ca trustpoint <ca-name>
```

4. To configure the LDAP parameters, enter:

```
crl query <url-with-ldap://>
```

**Table 113: Variable definitions**

Variable	Value
<url-with-ldap://>	Specifies a complete URL (with ldap://)

## Requesting a CRL from the CA

Request the Certificate revocation list (CRL) from the CA.

If you have configured the LDAP client parameters (`crl query <url-with-ldap://>`) the LDAP client is used to obtain the CRL. The LDAP client supports periodic downloads of the CRL using the “next update” time as specified by the CA in the CRL.

If the LDAP client parameters are not configured, and the SCEP URL is configured, the CRL is obtained using the SCEP client. However, the SCEP client does not support the periodic download of CRLs.

If the enrollment mode is set to terminal, the system prompts you to paste the obtained CRL.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the CRL configuration, enter:

```
ca crl
```

4. To request the CRL from the CA, enter:

```
request <ca-server-name>
```

---

## Configuring OCSP

Configure OCSP parameters. This specifies the URL of an OCSP server so that certificate status can be verified in real time during the IKE negotiation. To enable OCSP for a particular policy, you must use the `ike policy ocsp` command.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify the CA trustpoint to configure, enter:  
`ca trustpoint <ca-name>`
4. To configure the OCSP parameters, enter:  
`ocsp {nonce | signature | url}`

---

## Configuring IPsec VPN Support without Firewall

Disable the firewall globally to allow IPsec VPN to be configured without a firewall.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To disable the firewall to allow VPN configuration without the firewall, enter:  
`system security firewall-disable`

---

## Displaying IPsec VPN configurations

---

### Displaying certificates

To display certificates, enter:

```
show crypto ca certificates {<ca-name> | all} [detail]
```

---

### Displaying CRL

To display the CRL, enter:

```
show crypto ca crl {<ca-name> | all} [detail]
```

---

### Displaying trustpoint

To display trustpoint information, enter:

```
show crypto ca trustpoint {<ca-name> | all} [detail]
```

---

### Displaying IKE policies

To display IKE policies, enter:

```
show crypto ike policy {<policy-name> | all} [proposal-priority  
<1-5>] [detail]
```

---

### Displaying IKE SA

To display IKE SA information, enter:

```
show crypto ike sa {<policy-name> | all} [detail]
```

---

## Displaying IPsec policies

To display IPsec policies, enter:

```
show crypto contivity-iras ipsec policy {<policy-name> | all}
[proposal-priority <1-5>] [detail]
```

**Table 114: Variable definitions**

Variable	Value
<policy-name>	Identifies the IPsec policy name.
proposal-priority <1-5>	Identifies the IPsec proposal priority number. Values range from 1 to 5.

---

## Displaying IPsec SA

To display IPsec SA information, enter:

```
show crypto ipsec sa {<policy-name> | all} [detail]
```

---

## Displaying remote access IKE policies

To display remote access IKE policies, enter:

```
show crypto dynamic ike policy {<policy-name> | all} [proposal-
priority <1-5>] [detail]
```

---

## Displaying remote access IPsec policies

To display remote access IPsec policies, enter:

```
show crypto dynamic ipsec policy {<policy-name> | all}  
[proposal-priority <1-5>] [detail]
```

---

## Displaying remote access VPN clients

To display remote access IPsec clients, enter:

```
show crypto dynamic clients
```

---

## Displaying status of interfaces as trusted or untrusted

To display the status of interfaces, whether trusted or untrusted, enter:

```
show crypto interfaces
```

---

## Displaying dead peer detection configuration

To display the dead peer detection configuration, enter:

```
show crypto keepalive
```

---

## Displaying PMTU information

To display PMTU information, enter:

```
show crypto pmtu {config | df-bit <interface-name>}
```

---

## Displaying IPsec statistics

To display IPsec-related statistics, enter:

```
show crypto statistics
```

---

## Displaying L2TP server configuration

To display L2TP server configuration, enter:



```
show interface l2tp-server
```

---

## Clearing IPsec configurations

---

### Deleting certificates

To delete the CA certificate, enter:

```
clear crypto ca certificates ca-name <ca-name> serialnumber  
<serial-number>
```

---

### Deleting CRL

To delete the CRL, enter:

```
clear crypto ca crl <ca-name>
```

---

### Deleting CA private key

To delete the private key, enter:

```
clear crypto ca key <key-id>
```

---

### Clearing IKE SA information

To clear IKE SA information, enter:

```
clear crypto ike sa {<policy-name> | all}
```

---

### Clearing IPsec SA information

To clear IPsec SA information, enter:

```
clear crypto ipsec sa {<policy-name> | all}
```

---

## Clearing IPsec statistics

To clear IPsec statistics, enter:

```
clear crypto statistics
```

# Chapter 14: Avaya VPN client configuration

This chapter describes the steps required to configure a VPN connection for Avaya VPN clients.

The following are high-level steps to create an Avaya VPN client connection:

1. Create the new IKE policy. See [Creating an IKE policy for Avaya VPN client](#) on page 252.
2. Configure the local address. See [Configuring the IKE policy local address for Avaya VPN client](#) on page 252.
3. Configure the remote ID. See [Configuring the IKE policy remote ID for Avaya VPN client](#) on page 253.

When you configure the remote ID, an IKE policy with default parameters is created. No other IKE configuration can be entered until you configure at least one remote ID.

4. Configure the mandatory client configuration parameters: address pool and private address. See [Configuring an address pool for mode configuration](#) on page 255 and [Configuring a private side address](#) on page 256.

## **Important:**

Although not strictly mandatory, you can be required to configure at least one DNS server. See [Configuring DNS server address for mode configuration](#) on page 256.

5. Configure optional IKE policy and IKE proposal parameters as desired. See [Configuring IKE mode for Avaya VPN client](#) on page 254 and [Configuring IKE proposal parameters for Avaya VPN client](#) on page 266.

The default properties will interoperate with Avaya VPN client.

6. Configure an IPsec policy. See [Creating an IPsec policy for Avaya VPN client](#) on page 270.
7. Specify the matching IP address on which to apply IPsec. See [Specifying the IP stream on which to apply IPsec](#) on page 192.

When you configure the matching IP address, an IPsec policy with default parameters is created. No other IPsec configuration can be entered until the matching address is entered.

8. Optionally, configure additional IPsec parameters as desired. See [Configuring IPsec proposal parameters for Avaya VPN client](#) on page 270.

The default assumptions will interoperate with Avaya VPN client.

9. Enable the IPsec policy . See [Enabling or disabling the IPsec policy entry](#) on page 189.

In addition to the preceding steps, you must also perform the following configurations for interoperability with the firewall:

1. Configure at least one trusted interface and one untrusted interface.
2. Configure an IP route default for the destination addresses specified in the IPsec policies.

For traffic to be forwarded through the VPN, a route to the peer must be available. As you cannot know the address of the peers, you must assume that they all come from the default

route out to the internet. Even though the application traffic, matching the IPsec policy, is getting tunneled, the built-in firewall uses the IP route to cross check whether the router is expected to handle this traffic at all.

3. Configure three inbound firewall policies in the internet zone for IKE negotiation. For example:  
`policy 100 in permit service ike self policy 101 in permit service ike-nat self policy 102 in permit service contivity self`
4. If you are configuring a management tunnel, configure inbound firewall policies in the internet map for the required services (telnet, icmp, and so on.)
5. 5. If you are configuring a transit tunnel, configure inbound firewall policies in the appropriate map (for example, corp) for the required services.

---

## Configuring IKE policy parameters for Avaya VPN client

The following sections describe procedures for configuring IKE policy parameters for Avaya VPN client connections.

---

### Creating an IKE policy for Avaya VPN client

Create a dynamic IKE policy for Avaya VPN client.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To create a remote access IKE policy, enter:  
`ike policy <policy-name>`

---

### Configuring the IKE policy local address for Avaya VPN client

Configure the local address to be used in IKE negotiations.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To configure the local address, enter:

```
local-address <A.B.C.D>
```

## Configuring the IKE policy remote ID for Avaya VPN client

Configure the remote ID to specify the IPsec peer that participates in the IKE negotiation.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To configure the remote ID, enter:

```
remote-id {user-name "<user-name>" | group-name "<group-name>" | der-encoded-dn "<name>"} [password <password>]
```

### Important:

All remote IDs in a single IKE policy must be the same type (user, group, or der).

**Table 115: Variable definitions**

Variable	Value
user-name "<user-name>"	Specifies a user name. The value must be specified within quotation marks.
group-name "<group-name>"	Specifies a group name. The value must be specified within quotation marks.

Variable	Value
	The same value of group-name must be configured as the 'class-attribute' on IDE (Radius server) but without the quotation marks.
der-encoded-dn "<name>"	Specifies the x.509 (subject name) distinguished name. The value must be specified within quotation marks.
password <password>	Specifies the user or group password.

## Configuring IKE mode for Avaya VPN client

Configure the IKE mode for the remote access policy.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To configure the IKE mode, enter:  
`mode {main | aggressive}`

**Table 116:**

Variable	Value
main	Specifies use of full negotiation to establish a security association. Main mode provides identity protection.
aggressive	Specifies use of a faster phase 1 establishment mode. Aggressive mode does not provide identity protection. (This is the default mode.)

---

## Configuring mode configuration client parameters for Avaya VPN client

The following sections describe procedures for configuring mode-config client parameters for Avaya VPN client connections.

---

### Configuring an address pool for mode configuration

Use this procedure to configure an address pool. By default, an address pool is not configured.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure an address pool, enter:

```
[no] address-pool <pool-no> <start-address> <end-address>
```

**Table 117: Variable definitions**

Variable	Value
[no]	Removes the specified address pool configuration.
<pool-no>	Specifies a number for the address pool. Values range from 1 to 4.
<start-address>	Specifies the start IP address for the address pool.

Variable	Value
<code>&lt;end-address&gt;</code>	Specifies the end IP address for the address pool.

---

## Configuring a private side address

Use this procedure to configure a private side address for the Avaya VPN client connection. By default, a private side address is not configured.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To specify client configuration, enter:  
`client configuration`
6. To configure a private side address, enter:  
`[no] private-side-address <private-side-address>`

**Table 118: Variable definitions**

Variable	Value
<code>[no]</code>	Removes the specified private-side address configuration.
<code>&lt;private-side-address&gt;</code>	Specifies a crypto trusted IP address on this device. The client tests reachability to this address.

---

## Configuring DNS server address for mode configuration

Use this procedure to configure a DNS server address for the Avaya VPN client connection. By default, a DNS server address is not configured.



## Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To specify client configuration, enter:  
`client configuration`
6. To configure a DNS server address, enter:  
`[no] dns-server <primary-dns> <secondary-dns>`

**Table 119: Variable definitions**

Variable	Value
[no]	Removes the specified DNS server address.
<primary-dns>	Specifies the primary DNS server address.
<secondary-dns>	Specifies the secondary DNS server address.

## Configuring a WINS server address for mode configuration

Use this procedure to configure a WINS server address for the Avaya VPN client connection. By default, a WINS server is not configured.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure the WINS server address, enter:

```
[no] wins-server <primary-wins> <secondary-wins>
```

**Table 120: Variable definitions**

Variable	Value
[no]	Removes the specified WINS server address.
<primary-wins>	Specifies the primary WINS server address.
<secondary-wins>	Specifies the secondary WINS server address.

## Configuring split tunnel parameters

Use this procedure to configure split tunnel parameters. By default, split tunneling is disabled.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To specify split-tunnel configuration, enter:

```
split-tunnel
```

7. To specify the networks to configure for split-tunneling, enter:

```
[no] network <network> <prefix>
```

8. To specify the split-tunneling mode to use for the specified networks, enter:

```
[no] mode {disable | enabled | inverse | local-inverse}
```

**Table 121: Variable definitions**

Variable	Value
[no]	Removes the specified split-tunneling configuration.
disabled	Disables split tunneling. The Avaya VPN client routes all traffic through the tunnel. Disabled is the default value.
enabled	Configures the Avaya VPN client to route networks specified by the network command through the tunnel and other networks in the clear.
inverse	Configures the Avaya VPN client to route networks specified by the network command in the clear and other networks through the tunnel.
local-inverse	Configures the Avaya VPN client to route any networks which are locally connected to the client in the clear and route other networks through the tunnel.

## Configuring the client domain name

Use this procedure to configure the client domain name for the Avaya VPN client connection. By default, the client domain name is not configured.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure client domain name, enter:

```
[no] client-domain-name <domain-name>
```

**Table 122: Variable definitions**

Variable	Value
[no]	Removes the specified client domain name.
<domain-name>	Specifies the client domain name. The client assumes this value as its domain name suffix.

## Configuring whether the client can store username and password

Use this procedure to configure whether the Avaya VPN client can store username and password information. By default, this parameter is enabled.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure the client-may-store parameter, enter:

```
[no] client-may-store-password
```

**Table 123: Variable definitions**

Variable	Value
[no]	Disables storage of username and password on the Avaya VPN client.

## Configuring the client screen saver wait time

Use this procedure to configure the maximum allowed wait time, in minutes, to which the password-protected screen saver on the client host must be configured. If the wait-time exceeds this value, the host cannot activate the VPN connection. By default, the wait-time is not specified.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure client screen saver wait time, enter:

```
[no] client-screen-saver <minutes>
```

**Table 124: Variable definitions**

Variable	Value
[no]	Removes the specified client screen saver configuration.
<minutes>	Specifies the maximum number of minutes to which the password protected screen saver on the client host must be set before activation. The default value is 0, which indicates off or infinite time .

## Configuring the client banner text

Use this procedure to configure a banner to display on the Avaya VPN client. By default, banner text is not configured. When banner text is not configured, the system attempts to extract banner text from the following location: /cf0/contivityBanner.txt.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To specify client configuration, enter:  
`client configuration`
6. To configure the banner text, enter:  
`[no] banner-text "<banner-text>"`

**Table 125: Variable definitions**

Variable	Value
[no]	Removes the specified banner text.
<banner-text>	Specifies the Avaya VPN client banner up to 200 characters, delimited by quotation marks (" ").

---

**Enabling or disabling the client banner**

Use this procedure to enable or disable the banner for the Avaya VPN client. By default, the client banner is disabled.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To enable or disable the banner, enter:

```
[no] banner-enable
```

**Table 126: Variable definitions**

Variable	Value
[no]	Disables the client banner.

## Configuring failover for the Secure Router

Use this procedure to configure up to three IP addresses of failover for SR2330/4134. By default, no failover IP addresses are specified.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure the failover server addresses, enter:

```
[no] failover-list <failover-ip-1> <failover-ip-2>  
<failover-ip-3>
```

**Table 127: Variable definitions**

Variable	Value
[no]	Removes the specified failover for SR2330/4134.
<failover-ip-1>	Specifies the IP address of a primary failover for SR2330/4134.

Variable	Value
<failover-ip-2>	Specifies the IP address of a secondary failover for SR2330/4134.
<failover-ip-3>	Specifies the IP address of a tertiary failover for SR2330/4134.

## Configuring keepalive behavior

Use this procedure to configure keepalive behavior. By default, keepalives are disabled.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To specify client configuration, enter:  
`client configuration`
6. To specify keepalive configuration, enter:  
`keepalive`
7. To configure the keepalive interval, enter:  
`interval <interval>`
8. To configure the maximum retransmission value, enter:  
`retransmissions <retransmissions>`
9. To enable the keepalives, enter:  
`[no] enable`

**Table 128: Variable definitions**

Variable	Value
<interval>	Specifies the keepalive time interval. The default value is 0.



Variable	Value
<retransmissions>	Specifies the maximum number of retransmissions. The default value is 0.
[no]	Disables keepalives.

## Configuring NAT keepalives

Use this procedure to configure the frequency at which the Avaya VPN server sends Network Address Translation (NAT) keepalives when a NAT is detected. By default, NAT keepalives are disabled. Keepalive must first be enabled in order to enable NAT keepalive. To configure keepalive, see [Configuring keepalive behavior](#) on page 264.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify client configuration, enter:

```
client configuration
```

6. To configure NAT keepalives, enter:

```
[no] nat-keepalive <seconds>
```

**Table 129: Variable definitions**

Variable	Value
[no]	Removes the specified NAT keepalive configuration.
<seconds>	Specifies the NAT keepalive interval. Values range from 20 to 120 seconds.

---

## Configuring IKE proposal parameters for Avaya VPN client

The following sections describe procedures for configuring IKE proposal parameters for Avaya VPN client connections.

---

### Configuring an IKE proposal for Avaya VPN client

Configure an IKE proposal for Avaya VPN client.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To create the IKE proposal, enter:  
`proposal <1-5>`

**Table 130: Variable definitions**

Variable	Value
< 1-5>	Specifies the proposal value. Only one proposal is allowed in aggressive mode.

---

### Configuring authentication method for IKE proposal for Avaya VPN client

Configure the IKE proposal authentication method for Avaya VPN client.

#### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify the IKE proposal, enter:

```
proposal <1-5>
```

6. To configure the remote ID, enter:

```
authentication-method {pre-shared-key | rsa-signature}
```

**Table 131: Variable definitions**

Variable	Value
pre-shared-key	Specifies authentication using a pre-shared key, derived out of band.
rsa-signature	Specifies authentication using RSA Signature.

## Configuring DH group for IKE proposal for Avaya VPN client

Configure the DH group for the IKE proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name> <peer>
```

5. To specify the IKE proposal, enter:

```
proposal <1-5>
```

6. To configure DH group for IKE proposal, enter:

```
dh-group {group1 | group2 | group5}
```

**Table 132: Variable definitions**

Variable	Value
group1	768-bit. RFC 2409.
group2	1024-bit. RFC 2409.
group5	1536-bit. RFC2409. This is the highest level of security supported by the Avaya Secure Router 2330/4134 and requires more processing time than group 1 and group 2.

## Configuring encryption algorithm for IKE proposal for Avaya VPN client

Configure the encryption algorithm for the IKE proposal.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IKE policy to configure, enter:

```
ike policy <policy-name>
```

5. To specify the IKE proposal, enter:

```
proposal <1-5>
```

6. To configure the encryption algorithm for IKE, enter:

```
encryption-algorithm {des-cbc | 3des-cbc | aes128-cbc |  
aes192-cbc | aes256-cbc}
```

**Table 133: Variable definitions**

Variable	Value
des-cbc	Specifies DES-CBC encryption.
3des-cbc	Specifies 3DES-CBC encryption.

Variable	Value
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.

## Configuring IKE hash algorithm for Avaya VPN client

Configure the IKE authentication algorithm for a given IKE proposal.

### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IKE policy to configure, enter:  
`ike policy <policy-name>`
5. To specify the IKE proposal, enter:  
`proposal <1-5>`
6. To configure the hash algorithm for IKE, enter:  
`hash-algorithm {md5|sha1}`

**Table 134: Variable definitions**

Variable	Value
md5	A 128-bit message digest-RFC 1321.
sha1	Secure Hash Standard: A 160-bit message digest-NIST,FIPS PUB 180-1.

---

## Configuring IPsec proposal parameters for Avaya VPN client

The following sections describe procedures for configuring IPsec proposal parameters for Avaya VPN client connections.

---

### Creating an IPsec policy for Avaya VPN client

Create an IPsec policy for an IPsec SA.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To create a remote access IPsec policy, enter:  
`ipsec policy <name>`

---

### Configuring encryption algorithm for IPsec proposal for Avaya VPN client

Configure the encryption algorithm for the remote access IPsec proposal.

#### Procedure steps

1. To enter the configuration mode, enter:  
`configure terminal`
2. To specify crypto configuration for IPsec and IKE, enter:  
`crypto`
3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:  
`contivity-iras`
4. To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name>
```

- To specify the IPsec proposal template to configure, enter:

```
proposal <1-5>
```

- To configure the encryption algorithm, enter:

```
encryption-algorithm { 3des-cbc | aes128-cbc | aes192-cbc |  
aes256-cbc }
```

**Table 135: Variable definitions**

Variable	Value
3des-cbc	Specifies 3DES-CBC encryption.
aes128cbc	Specifies AES-CBC encryption, with 128-bit key length.
aes192cbc	Specifies AES-CBC encryption, with 192-bit key length.
aes256cbc	Specifies AES-CBC encryption, with 256-bit key length.

## Configuring the hash algorithm for IPsec proposal for Avaya VPN client

Configure the hash algorithm for the IPsec proposal.

### Procedure steps

- To enter the configuration mode, enter:

```
configure terminal
```

- To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

- To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

- To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name>
```

- To specify the IPsec proposal template to configure, enter:

```
proposal <1-5>
```

- To configure the hash algorithm, enter:

```
hash-algorithm {md5-hmac | sha1-hmac | null}
```

**Table 136: Variable definitions**

Variable	Value
md5	Specifies a 128-bit message digest-RFC 1321.
sha1	Specifies a secure Hash Standard: A 160-bit message digest-NIST,FIPS PUB 180-1.
null	No authentication.

## Configuring lifetime for IPsec proposal for Avaya VPN client

Configure the lifetime of the IPsec SA.

When the SA expires, it is either replaced by a new negotiated SA or terminated.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify remote-access IKE policy configuration for Avaya VPN clients, enter:

```
contivity-iras
```

4. To specify the remote access IPsec policy to configure, enter:

```
ipsec policy <name>
```

5. To specify the IPsec proposal template to configure, enter:

```
proposal <1-5>
```

6. To configure lifetime, enter:

```
lifetime {kilobytes <300-4194303> | seconds <300-864000>}
```

**Table 137: Variable definitions**

Variable	Value
kilobytes <0-4194303>	Specifies the lifetime in kilobytes. The default is 0 (unlimited).
seconds <100-864000>	Specifies the lifetime in seconds. The default value is 3600 seconds.



---

## Displaying Avaya VPN client configuration

The following sections describe procedures to display the Avaya VPN client configuration.

---

### Displaying IKE policies

To display IKE policies, enter:

```
show crypto contivity-iras ike policy {<policy-name> | all}  
[proposal-priority <1-5>] [detail]
```

**Table 138: Variable definitions**

Variable	Value
<policy-name>	Identifies the IKE policy name.
proposal-priority <1-5>	Identifies the IKE proposal priority number. Values range from 1 to 5.

---

### Displaying IPsec policies

To display IPsec policies, enter:

```
show crypto contivity-iras ipsec policy {<policy-name> | all}  
[proposal-priority <1-5>] [detail]
```

**Table 139: Variable definitions**

Variable	Value
<policy-name>	Identifies the IPsec policy name.
proposal-priority <1-5>	Identifies the IPsec proposal priority number. Values range from 1 to 5.

---

### Displaying Avaya VPN clients

To display remote access IPsec clients, enter:

```
show crypto clients contivity
```



# Chapter 15: GRE and IPIP tunnel configuration

This chapter describes the steps required to configure GRE and IPIP tunnel.

The following are high-level steps for tunnel configuration:

1. Create the tunnel.
2. Configure the tunnel mode.
3. Configure an IP address for the tunnel
4. Configure the tunnel source
5. Configure the tunnel destination.
6. Configure optional parameters.
7. Configure a static or dynamic IP route to reach the tunnel end point.

---

## Configuring a tunnel

To configure a tunnel, you must perform the following five procedures.

---

### Creating a tunnel

Create a tunnel.

#### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the name of the tunnel to create, enter:  

```
interface tunnel <tunnel-name>
```

---

### Configuring tunnel encapsulation mode

Configure the tunnel encapsulation mode. The default is GRE.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the tunnel to configure, enter:  
`interface tunnel <tunnel-name>`
3. To configure tunnel mode, enter:  
`tunnel mode {gre | ipip | ipv6ip [6to4]}`

**Table 140: Variable definitions**

Variable	Value
gre	Specifies GRE encapsulation.
ipip	Specifies IP over IP encapsulation.
ipv6ip [6to4]	Specifies IPv6 over IP encapsulation. If the 6to4 option is included, automatic IPv6 tunneling is used.

---

## Configuring an IP address for the tunnel

Configure the tunnel IP address.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the tunnel to configure, enter:  
`interface tunnel <tunnel-name>`
3. To configure the IP address for the tunnel, enter:  
`[ip address <ipv4-address> <subnet-mask>] | [ipv6 <ipv6-address>]`

**Table 141: Variable definitions**

Variable	Value
<ipv4-address> <subnet-mask>	Specifies the IPv4 address and subnet mask.
<ipv6-address>	Specifies the IPv6 prefix address. You can also enter the IPv6 prefix name.

## Configuring tunnel source

Use this procedure to assign a local interface to a tunnel source.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel_name>
```

3. To configure tunnel source, enter:

```
[no] tunnel source <A.B.C.D> | <interface_name> |  
<bundlename:pvc>
```

**Table 142: Variable definitions**

Variable	Value
<A.B.C.D>	Specifies the tunnel source IP address for a local interface.
<bundlename:pvc>	Specifies the bundle interface name as the tunnel source.
<interface_name>	Specifies the physical Ethernet interface name as the tunnel source.
[no]	Removes the local interface from the tunnel source.
<tunnel_name>	Specifies the interface tunnel identifier to a maximum of 8 characters.

## Configuring tunnel destination

Configure the IP address for the tunnel destination. This is not required for 6to4 tunnels.

The tunnel destination cannot be a point-to-point interface peer. It should be reachable through a physical interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel destination, enter:

```
[no] tunnel destination <A.B.C.D>
```

## Configuring TCP MSS on a GRE/IP/IP tunnel interface

Use this procedure to configure the TCP MSS value on a GRE/IP/IP tunnel interface.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To select the tunnel, enter:

```
interface tunnel <tunnel-name>
```

3. To specify the IP address for the tunnel, enter:

```
[ip address <ipv4-address> <subnet-mask>] | [ipv6 <ipv6-address>]
```

4. To specify the source address of the tunnel, enter:

```
tunnel source <A.B.C.D>
```

5. To specify the destination address of the tunnel, enter:

```
tunnel destination <A.B.C.D>
```

6. To specify the TCP MSS of the tunnel, enter:

```
ip tcp-mss <value>
```

7. To exit the tunnel configuration mode, enter:

```
exit
```

**Table 143: Variable definitions**

Variable	Value
<ipv4-address> <subnet-mask>	Specifies the IPv4 network address and subnet mask.
<ipv6-address>	Specifies the IPv6 network address.
ip tcp-mss <value>	Specifies the TCP MSS clamping value for the tunnel. Values range from 536 to 9176.
<tunnel-name>	Specifies the identifier for the tunnel interface.
tunnel source <A.B.C.D>	Specifies the tunnel source IP address.
tunnel destination <A.B.C.D>	Specifies the tunnel destination IP address.

---

## Configuring GRE tunnel parameters

---

### Configuring keepalive for GRE tunnels

Enable keepalive packets to keep track of the tunnel end points. The router sends a keepalive at every configured interval. If no response is received after the configured number of retries, the tunnel is brought down. You can only configure keepalive on GRE tunnels.

By default GRE keepalives are disabled.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To specify the keepalive for the tunnel, enter:

```
keepalive [interval <0-120>] [retries <1-16>]
```

**Table 144: Variable definitions**

Variable	Value
[interval <0-120>]	Specifies the keepalive interval in seconds. Default is 10. A value of 0 specifies no keepalives.
[retries <1-16>]	Specifies the number of retries. If there is no response after the configured retry value, the tunnel is brought down. Default value is 3.

---

### Configuring checksum for GRE tunnels

Enable end-to-end checksums to force the router to drop any corrupted packets. You can only configure checksum on GRE tunnels.

By default, checksums are disabled.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To enable or disable checksum for tunnel packets, enter:

```
[no] tunnel checksum
```

---

## Configuring tunnel key for GRE tunnels

Configure an ID key for a tunnel interface. This key must be set to the same value on the tunnel endpoints. The key field is used for identifying an individual traffic flow within a tunnel. Tunnel ID keys can be used as a form of weak security to prevent misconfiguration or injection of packets from a foreign source. However this is not a reliable security option.

You can only configure tunnel key on GRE tunnels. By default, no key is configured.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel key, enter:

```
[no] tunnel key <0-4294967295>
```

**Table 145: Variable definitions**

Variable	Value
<0-4294967295>	Specifies the key value.

---

## Configuring tunnel sequencing

Configure sequencing to enable dropping of out-of-order packets on the tunnel.

The default mode for this feature is disabled. This feature is available only in the GRE mode.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:



```
interface tunnel <tunnel-name>
```

3. To configure tunnel sequence, enter:

```
[no] tunnel sequence
```

---

## Configuring tunnel parameters

---

### Configuring path MTU discovery for tunnel packets

Enable or disable path MTU discovery on the tunnel. The default mode for this feature is disabled.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To enable or disable path MTU discovery, enter:

```
[no] tunnel path_mtu_discovery
```

---

### Configuring the tunnel as an untrusted interface for IPsec protection

Configure the tunnel as untrusted for the purpose of IPsec protection.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel protection, enter:

```
crypto {trusted | untrusted}
```

**Table 146: Variable definitions**

Variable	Value
trusted	Specifies that the tunnel is part of a trusted network.
untrusted	Specifies that the tunnel is part of an untrusted network.

---

## Configuring tunnel protection with IPsec

This command associates a tunnel with an IPsec profile. IPsec automatically derives the IPsec peer and proxy information from the configured tunnel source and tunnel destination values. Therefore, you do not need to create an IPsec policy to define peer and match addresses.

IPsec creates a default IKE policy and transport-mode IPsec policy for the tunnel endpoint.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel protection, enter:

```
[no] tunnel protection <ipsec-policy> <key-value>
```

**Table 147: Variable definitions**

Variable	Value
<ipsec-policy>	Specifies the name of the IPsec policy. Enter a word of no more than eight characters.
<key-value>	Specifies the IPsec policy key value. Enter a string of no more than 49 characters.

---

## Configuring tunnel ToS

Configure the Type of Service (ToS) value for the tunnel interface. If not specified, the ToS value of the inner IP header is copied to the outer IP header.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel ToS, enter:

```
[no] tunnel tos <0-255>
```

**Table 148: Variable definitions**

Variable	Value
<0-255>	The ToS for the tunnel.
[no]	Sets the ToS value to the default, which is 0 (no ToS).

---

## Configuring tunnel TTL

Configure the time-to-live (TTL) value for the tunnel interface. The default value is 30 seconds.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel TTL, enter:

```
[no] tunnel ttl <1-255>
```

---

## Shutting down a tunnel

Shut down the tunnel interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the tunnel to configure, enter:

```
interface tunnel <tunnel-name>
```

3. To configure tunnel TTL, enter:

```
[no] shutdown
```

---

## Configuring VLAN parameters for the tunnel

Configure switchport, spanning-tree, and gvrp parameters on the tunnel to provide VLAN support across the tunnel. For more details on these parameters, see *Secure Router 2330/4134 Configuration—Layer 2 Ethernet, NN47263-501*.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the tunnel to configure, enter:  
`interface tunnel <tunnel-name>`
3. To configure VLAN parameters, enter:  
`switchport`
4. To configure Spanning Tree parameters, enter:  
`spanning-tree`
5. To configure GVRP parameters, enter:  
`gvrp`

---

## Configuring PCAP over GRE

Use this following procedure to configure a packet capture session for a GRE tunnel interface.

### Procedure steps

1. To specify the PCAP name to configure, enter:  
`debug pcap capture <capture-name>`
2. To assign the PCAP name to the GRE tunnel interface, enter:  
`attach tunnel <tunnel-name>`

---

## Displaying tunnel information

To display tunnel configuration information, enter:

```
show interface [tunnel <tunnel-name> | tunnels]
```

---

## Clearing tunnel counters

To clear tunnel counter information, enter:

```
clear interface [tunnel <tunnel-name> | tunnels]
```



# Chapter 16: PPPoE client configuration

This chapter describes the steps required to configure a PPPoE client.

The following are high-level steps for PPPoE client configuration:

1. Create PPPoE interface.
2. Configure IP address.
3. Configure Ethernet interface.
4. Configure authentication used.

---

## Creating a PPPoE interface

Create a new PPPoE interface.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the PPPoE interface to create, enter:  
`interface virtual-access <pppoe-interface>`

---

## Configuring IP address for PPPoE interface

Configure the IP address for the PPPoE interface.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the name of the PPPoE interface to configure, enter:  
`interface virtual-access <pppoe-interface>`
3. To specify the IP address for this interface, enter:  
`{address <ip-address/mask> | negotiated }`

**Table 149: Variable definitions**

Variable	Value
address <ip-address/mask>	Specifies the IP address for the interface.
negotiated	Specifies that the IP address for the interface is obtained via PPP IPCP (IP Control Protocol) address negotiation with the PPPoE server.

---

## Configuring PPPoE tunneling protocol

Configure the PPPoE tunneling protocol. The only available option in this release is client mode (the default).

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the name of the PPPoE interface to configure, enter:  

```
interface virtual-access <pppoe-interface>
```
3. To specify the IP address for this interface, enter:  

```
protocol pppoe pppoe-mode [client]
```

---

## Configuring PPPoE Ethernet interface

Configure the Ethernet interface on which PPPoE is enabled.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify the name of the PPPoE interface to configure, enter:  

```
interface virtual-access <pppoe-interface>
```
3. To specify the Ethernet interface to use for PPPoE, enter:  

```
pppoe ethernet <slot/port>
```



---

## Configuring PPP authentication method and parameters

Configure the PPP authentication method and related parameters for the PPPoE connection.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the PPPoE interface to configure, enter:

```
interface virtual-access <pppoe-interface>
```

3. To configure the authentication method for the PPPoE interface, enter:

```
ppp authentication {pap | chap | pap_chap | none} [sent-username <username>] [password <password>]
```

**Table 150: Variable definitions**

Variable	Value
[pap   chap   pap_chap   none]	Specifies the authentication method: one of PAP, CHAP, either, or none.
[sent-username <username>]	Specifies the local username to be authenticated (max is 64 characters). This parameter is required for all authentication types, unless <b>none</b> is specified.
[password <password>]	Specifies the local password to be authenticated (max is 64 characters). This parameter is required for all authentication types, unless <b>none</b> is specified.

---

## Configuring PPPoE access concentrator

Configure a specific PPPoE server.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the PPPoE interface to configure, enter:

```
interface virtual-access <pppoe-interface>
```

3. To configure access concentrator for PPPoE, enter:

```
pppoe ac-name <ac-name>
```

**Table 151: Variable definitions**

Variable	Value
<ac-name>	Specifies the PPPoE access concentrator name (max 32 characters).

---

## Configuring PPP keepalive

Configure the PPP keepalive interval in seconds. This value specifies the amount of time PPP stays up when there is no traffic.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the name of the PPPoE interface to configure, enter:

```
interface virtual-access <pppoe-interface>
```

3. To configure keepalive interval for the PPPoE interface, enter:

```
ppp keepalive interval <0-32767>
```

**Table 152: Variable definitions**

Variable	Value
<0-32767>	Specifies the amount of time in seconds that PPP stays up when there is no traffic. The default value is 10 seconds. A value of 0 disables keepalives.

---

## Displaying PPPoE client information

To display PPPoE client configuration information, enter:

```
show interface virtual-access <pppoe-interface>
```

# Chapter 17: Authentication, Authorization, and Accounting configuration

This chapter describes the steps required to enable and configure Authentication, Authorization, and Accounting (AAA).

---

## Enabling AAA

Enable or disable AAA on the router.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To enable and disable AAA, enter:

```
[no] aaa enable
```

**Table 153: Variable definitions**

Variable	Value
[no]	Disables AAA.

---

## Configuring AAA authentication

---

### Configuring AAA authentication login

Configure the login authentication methods.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the login authentication methods, enter:

```
aaa authentication login <login-list-name> <authentication-
methods>
```

**Table 154: Variable definitions**

Variable	Value
<login-list-name>	The name of the login authentication list, either a character string or the word <b>default</b> . If list_name is default, all interfaces use this method list without further configuration.
<authentication-methods>	A list of up to 3 authentication values, separated by slashes (/), indicating the order in which the methods are used for login on an interface. Possible values are tacacs, radius, local and none.

---

## Configuring AAA authentication protocol

Configure the login authentication protocols.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the login authentication protocols, enter:

```
aaa authentication protocol <protocols-list-name>
<protocols>
```

**Table 155: Variable definitions**

Variable	Value
<protocols-list-name>	The name of the protocol authentication list, either a character string or the word <b>default</b> . If list_name is default, all interfaces use this method list without further configuration.
<protocols>	A list of up to 3 protocol values, separated by slashes (/), indicating the order in which the protocols are used for login on an interface. Possible values are pap, chap and ascii.

---

## Applying AAA authentication to an interface

Apply AAA authentication to an interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the interface to configure, enter:

```
interface {bundle <bundle-name> | ethernet <slot/port> |  
console}
```

3. To apply AAA authentication to the interface, enter:

```
aaa authentication <login-list-name> <protocol-list-name>
```

---

## Configuring AAA authorization

---

### Configuring AAA authorization

Configure the authorization methods for accessing an interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the login authorization methods, enter:

```
aaa authorization commands <commands-list-name>  
<authorization-methods>
```

**Table 156: Variable definitions**

Variable	Value
<commands-list-name>	The name of the authorization list, either a character string or the word <b>default</b> . If list_name is default, all interfaces use this method list without further configuration.
<authorization-methods>	A list of up to 2 authorization values, separated by slashes (/), indicating the order in which the methods are used for

Variable	Value
	authorization on an interface. Possible values are tacacs, local and none.

---

## Applying AAA authorization to an interface

Apply AAA authorization to an interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the interface to configure, enter:

```
interface {bundle <bundle-name> | ethernet <slot/port> | console}
```

3. To apply AAA authorization to the interface, enter:

```
aaa authorization <commands-list-name>
```

---

## Configuring AAA accounting

---

### Configuring AAA accounting

Configure accounting on the router.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure accounting, enter:

```
aaa accounting [commands | network | exec | system ] <list-name> [start_stop | stop_only | wait_start | none]
```

**Table 157: Variable definitions**

Variable	Value
[commands   network   exec   system ] <list-name>	commands: configure the accounting for commands. network: configure the accounting for network usage. exec:

Variable	Value
	configure the accounting for EXEC sessions. system: configure the accounting for system events. <list-name> specifies the name of the accounting list, either a character string or the word <b>default</b> . If list_name is default, all interfaces use this method list without further configuration.
[start_stop   stop_only   wait_start   none]	The set of records sent for accounting are as follows: start_stop : START and STOP records are sent stop_only : only STOP records are sent wait_start : START and STOP records are sent. Service starts after ACK. none : No accounting

## Configuring AAA accounting update

Configure the network accounting update scheme.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the accounting update scheme, enter:

```
aaa accounting update {newinfo | periodic mins <1-5> }
```

**Table 158: Variable definitions**

Variable	Value
newinfo	Send UPDATE records when new info is available
periodic	Send UPDATE records periodically
mins <1-5>	Time in minutes if the scheme is periodic. Default is 5 mins.

## Applying AAA accounting to an interface

Apply AAA accounting to an interface.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the interface to configure, enter:

```
interface {bundle <bundle-name> | ethernet <slot/port> |  
console}
```

3. To apply AAA accounting to the interface, enter:

```
aaa accounting commands <commands-list-name> exec <exec-list-  
name> network <network-list-name> system <system-list-name>
```

---

## Configuring the AAA source address

Use the following procedure to configure the source address for AAA services.

### Procedure Steps

1. To configure source addresses for a service, enter Configuration Mode.

```
configuration terminal
```

2. To configure Radius or TACACS source addresses, enter the **aaa** command subtree.

```
aaa
```

3. Configure the source address.

```
source-address <[ip-address] | [interface-name]>
```

**Table 159: Variable definitions**

Variable	Value
[ip-address]	Specifies the source address by IP address.
[interface-name]	Specifies the source address by interface name.

---

## Configuring RADIUS primary and secondary servers

---

### Configuring RADIUS server port for accounting

Configure the port used by the RADIUS server for accounting.

#### Procedure steps

1. To enter configuration mode, enter:



```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS server accounting port, enter:

```
[no] acct_port <1-65535>
```

**Table 160: Variable definitions**

Variable	Value
<1-65535>	The accounting port on the RADIUS server. The default is port number 1813.

---

## Configuring RADIUS server port for authentication

Configure the port used by the RADIUS server for authentication.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS server authentication port, enter:

```
[no] auth_port <1-65535>
```

**Table 161: Variable definitions**

Variable	Value
<1-65535>	The authentication port on the RADIUS server. The default is port number 1812.

---

## Configuring the RADIUS server IP address

Configures the IP address of the specified RADIUS server. (A primary RADIUS server must be configured to enable RADIUS.)

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS server IP address, enter:

```
[no] ipaddress <A.B.C.D>
```

**Table 162: Variable definitions**

Variable	Value
<A.B.C.D>	The IP address of the specified RADIUS server.

---

## Configuring RADIUS client retries

Configure the number of client attempts to communicate with the RADIUS server.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS client retries, enter:

```
retries <1-5>
```

**Table 163: Variable definitions**

Variable	Value
<1-5>	The number of attempts to contact the server.

---

## Configure RADIUS shared secret key

Configure a secret key used by both the RADIUS client and server

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS shared secret key, enter:

```
shared_key <shared-key>
```

**Table 164: Variable definitions**

Variable	Value
<shared-key>	A string of length less than or equal to 48 characters.

---

## Configure RADIUS timeout

Configure the maximum wait time for a server response.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify configuration of the primary or secondary server, enter:

```
aaa radius [primary_server | secondary_server]
```

3. To configure the RADIUS timeout, enter:

```
time_out <1-100>
```

**Table 165: Variable definitions**

Variable	Value
1-100	The timeout in seconds. The maximum value is 100.

---

## Configuring RADIUS client source address

Configure the IP address of the RADIUS client. This address will be used for all packets sent from the RADIUS client to the RADIUS server.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the IP address of the RADIUS client, enter:

```
aaa radius src_address <A.B.C.D>
```

## Configuring TACACS accounting

Use the following procedure to configure TACACS accounting.

### Procedure steps

1. To configure TACACS accounting, enter Configuration Mode.

```
configure terminal
```

2. Enter the aaa command sub-tree.

```
aaa
```

3. Configure an access-list for commands.

```
accounting commands <listname|[default]> {start_stop|  
stop_only|wait-start}
```

4. Configure an access-list for a network.

```
accounting network <listname|[default]> {start_stop|  
stop_only|wait-start}
```

5. Exit back a level.

```
exit
```

6. Enter Interface Mode.

```
interface <interface>
```

7. Apply accounting to the interface.

```
aaa accounting {commands|network} <list>
```

**Table 166: Variable definitions**

Variable	Value
{commands networks}	Specifies the type of accounting to apply to the interface.
<interface>	Specifies the interface to work with.
<list>	Specifies the list to apply to the interface.
<listname>	Specifies the name of the accounting list. If list name is specified as "default", all interfaces use this list without further configuration.
{start_stop stop_only wait-start}	<ul style="list-style-type: none"> <li>• start_stop—Start and Stop records are sent.</li> <li>• stop_only—Only Stop records are sent.</li> <li>• wait-start—Start and Stop records are sent, but service starts after acknowledgement.</li> </ul>

---

## Configuring TACACS+ primary or secondary server IP address

Configure the IP address of the primary or secondary TACACS+ server.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To configure the IP address of the primary or secondary server, enter:

```
aaa tacacs [primary_server |secondary_server] <A.B.C.D>
```

---

## Configuring TACACS+ retries

Configure the number of client attempts to communicate with the TACACS+ server.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify TACACS+ configuration, enter:

```
aaa tacacs
```

3. To configure TACACS+ retries, enter:

```
[no] retries <1-5>
```

**Table 167: Variable definitions**

Variable	Value
<1-5>	The number of attempts to contact the server. The range is 1-5. The default is 2.

---

## Configuring TACACS+ server port

Configure the port used by the TACACS+ server.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify TACACS+ configuration, enter:  
`aaa tacacs`
3. To configure the TACACS+ server port, enter:  
`server_port <1-65535>`

**Table 168: Variable definitions**

Variable	Value
<1-65535>	The port on the TACACS+ server.

---

## Configuring TACACS+ shared encryption key

Configure a secret key used by both the TACACS+ client and server

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify TACACS+ configuration, enter:  
`aaa tacacs`
3. To configure the TACACS+ shared encryption key, enter:  
`[no] shared_key <encryption-key>`

**Table 169: Variable definitions**

Variable	Value
<encryption-key>	A string of length less than or equal to 8 characters.

---

## Configuring TACACS+ timeout

Configure the maximum wait time, in seconds, for a TACACS+ server response.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify TACACS+ configuration, enter:  
`aaa tacacs`
3. To configure the TACACS+ timeout in seconds, enter:  
`time_out <1-300>`

---

## Configuring 802.1x

Globally enable or disable 802.1x authentication. By default, 802.1x authentication is disabled.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To enable or disable 802.1x, enter:  
`dot1x [enable | disable]`

---

## Configuring 802.1x on an Ethernet interface

EAP IEEE 802.1x can be applied to Ethernet interfaces only.

---

### Enable 802.1x on the interface

Enable or disable 802.1x on the interface.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the Ethernet interface to configure, enter:  
`interface ethernet <slot/port>`
3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To enable or disable 802.1x on the interface, enter:

```
[no] dot1x-enable
```

---

## Configuring the maximum failed requests

Set the maximum of failed EAP requests sent to the supplicant. The default is 2.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the Ethernet interface to configure, enter:

```
interface ethernet <slot/port>
```

3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To set the maximum failed requests value, enter:

```
max-req <1-10>
```

---

## Configuring port control

Configure a forced 802.1x state for a port.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the Ethernet interface to configure, enter:

```
interface ethernet <slot/port>
```

3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To configure a forced 802.1x state for the port, enter:

```
[no] port-control {auto | force-authorized | force-unauthorized}
```



**Table 170: Variable definitions**

Variable	Value
auto	Enable authentication on a port.
force-authorized	Force a port to always be in an authorized state.
force-unauthorized	Force a port to always be in an unauthorized state.
no	Removes the port from 802.1x management.

---

## Configuring quiet period

Set the quiet-period time interval. The default is 60 seconds.

When the system cannot authenticate a client, the system remains idle for a quiet-period interval of time, then tries again. By administratively changing the quiet-period interval, by entering a lower number than the default, a faster response time can be provided.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the Ethernet interface to configure, enter:  
`interface ethernet <slot/port>`
3. To specify 802.1x configuration, enter:  
`dot1x`
4. To configure the quiet period, enter:  
`quiet-period <1-65535>`

---

## Enabling reauthentication

Enable or disable reauthentication on a port.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the Ethernet interface to configure, enter:  
`interface ethernet <slot/port>`
3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To enable or disable reauthentication, enter:

```
reauthentication {enable | disable}
```

---

## Configuring reauthorization period

Set the interval between reauthorization attempts. The default is 3600 seconds.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the Ethernet interface to configure, enter:

```
interface ethernet <slot/port>
```

3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To configure the reauthorization period in seconds, enter:

```
reauth-period <1-65535>
```

---

## Configuring authentication server response timeout

Set the authentication sever response timeout. The default is 30 seconds.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify the Ethernet interface to configure, enter:

```
interface ethernet <slot/port>
```

3. To specify 802.1x configuration, enter:

```
dot1x
```

4. To configure the authentication server timeout, in seconds, enter:

```
server-timeout <1-65535>
```

---

## Configuring supplicant response timeout

Set the supplicant response timeout. The default is 30 seconds.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify the Ethernet interface to configure, enter:  
`interface ethernet <slot/port>`
3. To specify 802.1x configuration, enter:  
`dot1x`
4. To configure the supplicant response timeout, in seconds, enter  
`supplicant-timeout <1-65535>`

---

## Displaying AAA information

---

### Displaying AAA accounting information

To display AAA accounting information, enter:

```
show aaa accounting [commands | exec | network | system |  
update]
```

---

### Displaying AAA authentication information

To display AAA authentication information, enter:

```
show aaa authentication [login | protocols]
```

---

### Displaying AAA authorization information

To display AAA authorization information, enter:

```
show aaa authorization commands
```

---

## Displaying AAA interface information

To display AAA interface information, enter:

```
show aaa interface {bundle <bundle-name> | ethernet <slot/port>
| console}
```

---

## Displaying AAA status

To display AAA status information, enter:

```
show aaa status
```

---

## Displaying RADIUS information

To display RADIUS information, enter:

```
show aaa radius
```

---

## Displaying TACACS+ information

To display TACACS+ information, enter:

```
show aaa tacacs
```

---

## Displaying 802.1x information

To display 802.1x information, enter:

```
show dot1x {detail | interface <if-name>| statistics}
```

---

## Clearing 802.1x statistics

To clear 802.1x statistics, enter:

```
clear dot1x statistics
```



# Chapter 18: SSH2 configuration

This chapter describes the steps required to configure SSH version 2.0 (SSH2).

---

## Configuring SSH2 server keys

---

### Generating SSH2 server keys

Generate host and user authentication keys.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 key configuration, enter:

```
ssh_keygen
```

3. Generate SSH2 server keys, enter:

```
generate {rsa | dsa} [outfile <filename>] [passphrase  
"<passphrase>"] [bits <512-2048>] [comment "<comment>"]
```

**Table 171: Variable definitions**

Variable	Value
{rsa   dsa}	Specifies the type of key to generate, either RSA or DSA.
[outfile <filename>]	Specifies the name of the files to contain the generated keys. The private key is stored in a file with the provided file name, while the public key is stored in a file with the same name with the extension .pub. The default file name is shrsakey for RSA keys, and shdsakey for DSA keys.
[passphrase "<passphrase>"]	Specifies the passphrase to encrypt the key file. The default is a null string. The passphrase variable must be specified within double quotes. This option should not be set when generating host keys.
[bits <512-2048>]	Specifies the length of the key, in bits. The default is 1024.

Variable	Value
[comment "<comment>"]	Specifies a string to identify the key. The comment variable must be specified within double quotes. The default is "user@hostname"

---

## Encrypting a private key file

Encrypt a private key file. The private key used for secure shell server should not be passphrase protected because you must be able to start the secure shell server without manual intervention. This command can be used to encrypt the private key with keys unique to the Router. The "no" form of the command can be used to decrypt the file.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 key configuration, enter:  
`ssh_keygen`
3. Encrypt the private key file, enter:  
`encrypt <private-key-filename>`

---

## Changing the passphrase used for encryption

Change the passphrase used to encrypt a key file.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 key configuration, enter:  
`ssh_keygen`
3. Change the passphrase, enter:  
`change "<old-passphrase>" "<new-passphrase>" <key-filename>`

**Table 172: Variable definitions**

Variable	Value
"<old-passphrase>"	The current passphrase. Must be specified in double quotes.



Variable	Value
"<new-passphrase>"	The new passphrase. Must be specified in double quotes.
<key-filename>	The name of the encrypted key file.

## Converting public key files to SSH format

Convert a public key file in OpenSSH format (the default) to a Secure Shell Standard format, or vice versa.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 key configuration, enter:

```
ssh_keygen
```

3. Change the passphrase, enter:

```
convert {secsh | openssh} <key-filename> newfile <new-key-filename>
```

**Table 173: Variable definitions**

Variable	Value
{secsh   openssh}	The type of conversion to perform: secsh: to convert from OpenSSH public key to SECSH public key openssh: to convert from unencrypted private or public SECSH key to OpenSSH.
<key-filename>	The name of the existing key file to convert.
<new-key-filename>	The name of the new key file to create. If this parameter is omitted, the output is displayed on the screen

## Generating a public key digest of a key file

Generate key digest of the key file. This is mainly used to compare key files.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 key configuration, enter:

```
ssh_keygen
```

3. Change the passphrase, enter:

```
digest <public-key-filename> [digest {fingerprint |  
bubblebabble}]
```

**Table 174: Variable definitions**

Variable	Value
<public-key-filename>	The name of the key file.
[digest {fingerprint   bubblebabble}]	The type of digest to generate, either fingerprint or bubblebabble. Default is fingerprint.

---

## Configuring SSH2 server parameters

---

### Configuring SSH2 authentication

Configure SSH2 user authentication.

#### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 server configuration, enter:

```
ssh_server
```

3. To configure SSH2 authentication, enter:

```
authentication {password | publickey}
```

**Table 175: Variable definitions**

Variable	Value
password	Specifies password based user authentication.
publickey	Specifies public key based user authentication.

---

### Configuring SSH2 authentication retries

Configure the number of authentication retries.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To configure the number of authentication retries, enter:  
`authRetries <1-5>`

**Table 176: Variable definitions**

Variable	Value
<1-5>	Specifies the number of authentication retries (default: 3).

---

## Configuring SSH encryption algorithms

Configure SSH encryption algorithms.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To configure the encryption algorithm, enter:  
`cipher {3descbc | blowfishcbc | aes128cbc | aes192cbc |  
aes256cbc}`

**Table 177: Variable definitions**

Variable	Value
3descbc	Specifies DES encryption.
blowfishcbc	Specifies blowfish encryption.
aes128cbc	Specifies AES, with 128-bit key length.
aes192cbc	Specifies AES, with 192-bit key length.
aes256cbc	Specifies AES, with 256-bit key length.

---

## Configuring SSH compression

Configure SSH compression.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To configure SSH compression, enter:  
`compression {none | zlib}`

**Table 178: Variable definitions**

Variable	Value
none	Specifies no compression.
zlib	Specifies zlib (LZ77) compression.

---

## Enabling and disabling SSH server

Enable or disable the SSH server.

### Procedure steps

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To enable or disable the SSH server, enter:  
`[no] enable`

---

## Specifying host key file for the SSH server

Specify the host key file for the SSH server.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To specify the host key file name, enter:  
`hostfile <key-filename>`

**Table 179: Variable definitions**

Variable	Value
<key-filename>	Specifies the host key file name. Default: shdsakey

---

## Enabling and disabling log events

Enable or disable log events.

**Procedure steps**

1. To enter configuration mode, enter:  
`configure terminal`
2. To specify SSH2 server configuration, enter:  
`ssh_server`
3. To enable or disable log events, enter:  
`[no] logevents`

**Table 180: Variable definitions**

Variable	Value
[no]	Disables log events.

---

## Configuring MAC algorithms

Configure MAC algorithms.

**Procedure steps**

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 server configuration, enter:

```
ssh_server
```

3. To configure the MAC algorithms, enter:

```
mac {hmacsha1 | hmacsha196 | hmacmd5 | hmacmd596}
```

**Table 181: Variable definitions**

Variable	Value
hmacsha1	hmac-sha1
hmacsha196	hmac-sha1-96
hmacmd5	hmac-md5
hmacmd596	hmac-md5-96

---

## Configuring SSH listen port

Configure the SSH listen port.

### Procedure steps

1. To enter configuration mode, enter:

```
configure terminal
```

2. To specify SSH2 server configuration, enter:

```
ssh_server
```

3. To configure the SSH listen port, enter:

```
port <512-65535>
```

**Table 182: Variable definitions**

Variable	Value
<512-65535>	Specifies the port value. Default: 22.

---

## Restoring default SSH parameter values

Restore the default SSH parameter values.

### Procedure steps

1. To enter configuration mode, enter:

- ```
configure terminal
```
2. To specify SSH2 server configuration, enter:  

```
ssh_server
```
  3. Restore default SSH parameter values, enter:  

```
restore
```

---

## Enabling and disabling SSH SFTP server

Enable or disable the SSH secure FTP (SFTP) server. By default, the SFTP server is disabled.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify SSH2 server configuration, enter:  

```
ssh_server
```
3. To enable or disable SSH SFTP server, enter:  

```
[no] sftpd
```

**Table 183: Variable definitions**

| Variable | Value                 |
|----------|-----------------------|
| no       | Disables SFTP server. |

---

## Configuring SSH session timeout

Configure the SSH session timeout.

### Procedure steps

1. To enter configuration mode, enter:  

```
configure terminal
```
2. To specify SSH2 server configuration, enter:  

```
ssh_server
```
3. To configure the SSH session timeout, enter:  

```
timeout <0-3600>
```

**Table 184: Variable definitions**

| Variable | Value                                          |
|----------|------------------------------------------------|
| 0-3600   | The default is 900 seconds. 0 means no timeout |

## Configuring an SFTP client

Use the following procedure to configure an SFTP client.

**Note:**

This feature supports only one SFTP client at a time, and it does not support IPv6 addresses.

The syntax for the new **sftp** command is:

```
sftp hostname <value> [cipher <value>] [mac <value>] [port <value>].
```

### Procedure

1. Enter the SFTP hostname:  

```
sftp hostname <value>
```
2. Configure the optional parameters, as needed.

## Variable definitions

Use the data in the following table to configure the **sftp** command.

| Variable         | Value                                                                                                                                                                                                                                                       |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| hostname <value> | This is a required parameter that specifies the name of the host. Enter the host name using either of the following formats: <ipAddress> or <username@ipAddress>. If you specify an IP address only, SFTP uses the login user name.<br><b>Default:</b> none |
| cipher <value>   | This is an optional parameter that specifies which encryption algorithm to use. The options are: <ul style="list-style-type: none"> <li>• none</li> <li>• des</li> <li>• blowfish</li> <li>• blowfish-cbc</li> </ul>                                        |



| Variable     | Value                                                                                                                                                                                                                                                         |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | <ul style="list-style-type: none"> <li>• 3des-cbc</li> <li>• aes128-cbc</li> <li>• aes192-cbc</li> <li>• aes256-cbc</li> </ul> <b>Default:</b> aes128-cbc                                                                                                     |
| mac <value>  | <p>This is an optional parameter that specifies which Hash-based Message Authentication Code (hmac) to use. The options are:</p> <p>hmac-sha1<br/>hmac-sha1-96<br/>hmac-md5<br/>hmac-md5-96<br/>hmac-ripemd160<br/>Optional<br/><b>Default:</b> hmac-sha1</p> |
| port <value> | <p>This is an optional parameter that specifies which port to connect to.<br/>The range is 1–65535.<br/><b>Default:</b> 22</p>                                                                                                                                |

---

## Displaying SSH server configuration

To display the SSH server configuration, enter:

```
show ip ssh config
```

---

## Displaying SSH server sessions

To display the SSH server sessions, enter:

```
show ip ssh {session <1-5> | sessions}
```

---

## Clearing SSH sessions

To clear SSH server sessions, enter:

```
clear ip ssh {session <1-5> | all}
```



# Chapter 19: Configuration examples

This chapter provides Avaya Secure Router 2330/4134 security configuration examples.

---

## Configuring an IPv4 packet filter

To configure an IPv4 packet filter named filterSet1, perform the following steps:

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the IPv4 packet filter, enter:

```
ip packet-filter filterSet1
add permit udp 10.0.0.0/24 any dport =53
add permit udp 10.0.0.0/24 any dport =138
add permit tcp 10.0.0.0/24 any flags established
add permit tcp 10.0.0.0/24 any flags syn log on
add permit ip 1.2.3.4/255.0.255.0 any
add permit ip 10.0.0.1/32 224.0.0.18/32
add permit ip any any proto 89
add deny ip any any log on
exit exit
show ip packet-filter filterSet1
```

IPv4 Filter Rule List : filterSet1

1. permit udp 10.0.0.0/24 any dport =53
2. permit udp 10.0.0.0/24 any dport =138
3. permit tcp 10.0.0.0/24 any flags established
4. permit tcp 10.0.0.0/24 any flags syn log on
5. permit ip 0.2.0.4/255.0.255.0 any
6. permit ip 10.0.0.1/32 224.0.0.18/32
7. permit 89 any any
8. deny ip any any log on

```
configure terminal
ip packet-filter filterSet1
insert 8 permit icmp any 10.0.0.1 log on
insert 9 permit icmp any 10.0.0.255 log on
exit
exit
show ip packet-filter filterSet1
```

IPv4 Filter Rule List : filterSet1

1. permit udp 10.0.0.0/24 any dport =53
2. permit udp 10.0.0.0/24 any dport =138
3. permit tcp 10.0.0.0/24 any flags established
4. permit tcp 10.0.0.0/24 any flags syn log on
5. permit ip 0.2.0.4/255.0.255.0 any
6. permit ip 10.0.0.1/32 224.0.0.18/32
7. permit 89 any any
8. permit icmp any 10.0.0.1/32 log on
9. permit icmp any 10.0.0.255/32 log on
10. deny ip any any log on

```
configure terminal
ip packet-filter filterSet1
delete 9
exit
exit
show ip packet-filter filterSet1
```

IPv4 Filter Rule List : filterSet1

1. permit udp 10.0.0.0/24 any dport =53
2. permit udp 10.0.0.0/24 any dport =138
3. permit tcp 10.0.0.0/24 any flags established
4. permit tcp 10.0.0.0/24 any flags syn log on
5. permit ip 0.2.0.4/255.0.255.0 any
6. permit ip 10.0.0.1/32 224.0.0.18/32
7. permit 89 any any
8. permit icmp any 10.0.0.1/32 log on
9. deny ip any any log on

---

## Configuring an IPv6 packet filter

To configure an IPv6 packet filter named filterSet1, perform the following steps:

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the IPv6 packet filter, enter:

```
ipv6 packet-filter filterSet1
add permit udp 0001::0/16 any dport =53
add permit udp 0001::0/16 any dport =138
add permit tcp 0001::0/16 any flags established
add permit tcp 0001::0/16 any flags syn log on
add permit tcp 0001::0/16 any
add permit ipv6 0001::0/16 0002::0/16
add permit ipv6 any any flowlabel 42
add permit ipv6 any any flowlabel 43 log on
add deny ipv6 any any routing on log on
add deny ipv6 any any log on
exit
exit
show ipv6 packet-filter filterSet1
```

```
IPv6 Filter Rule List : filterSet1
```

1. permit udp 1::/16 any dport =53
2. permit udp 1::/16 any dport =138
3. permit tcp 1::/16 any flags established
4. permit tcp 1::/16 any flags syn log on
5. permit tcp 1::/16 any
6. permit ipv6 1::/16 2::/16
7. permit ipv6 any any flowlabel 42
8. permit ipv6 any any flowlabel 43 log on
9. deny ipv6 any any routing onlog on
10. deny ipv6 any any log on

```
configure terminal
ipv6 packet-filter filterSet1
insert 9 permit ipv6 1::/64 any
exit
exit
show ipv6 packet-filter filterSet1
```

```
IPv6 Filter Rule List : filterSet1
```

1. permit udp 1::/16 any dport =53
2. permit udp 1::/16 any dport =138
3. permit tcp 1::/16 any flags established
4. permit tcp 1::/16 any flags syn log on
5. permit tcp 1::/16 any

6. permit ipv6 1::/16 2::/16
7. permit ipv6 any any flowlabel 42
8. permit ipv6 any any flowlabel 43 log on
9. permit ipv6 1::/64 any
10. deny ipv6 any any routing onlog on
11. deny ipv6 any any log on

```
configure terminal
ipv6 packet-filter filterSet1
delete 5
exit
exit
show ipv6 packet-filter filterSet1
```

IPv6 Filter Rule List : filterSet1

1. permit udp 1::/16 any dport =53
2. permit udp 1::/16 any dport =138
3. permit tcp 1::/16 any flags established
4. permit tcp 1::/16 any flags syn log on
5. permit ipv6 1::/16 2::/16
6. permit ipv6 any any flowlabel 42
7. permit ipv6 any any flowlabel 43 log on
8. permit ipv6 1::/64 any
9. deny ipv6 any any routing onlog on
10. deny ipv6 any any log on

---

## Configuring a MAC packet filter

To configure a MAC packet filter named filterSet1, perform the following steps:

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the MAC packet filter, enter:

```
mac packet-filter filterSet1
add permit any any vlan 20
add permit 1111.2222.0000 any smask 0000.0000.FFFF
add permit any any ethertype arp
exit
```

```
exit
show mac packet-filter filterSet1
```

Mac Filter Rule List : filterSet1

1. permit any any vlanid 20
2. permit any any
3. permit any any ethertype ARP

```
configure terminal
mac packet-filter filterSet1
add permit any any cos 7
add deny any any
exit
exit
show mac packet-filter filterSet1
```

Mac Filter Rule List : filterSet1

1. permit any any vlanid 20
2. permit any any
3. permit any any ethertype ARP
4. permit any any cos 7
5. deny any any

```
configure terminal
mac packet-filter filterSet1
delete 4
exit
exit
show mac packet-filter filterSet1
```

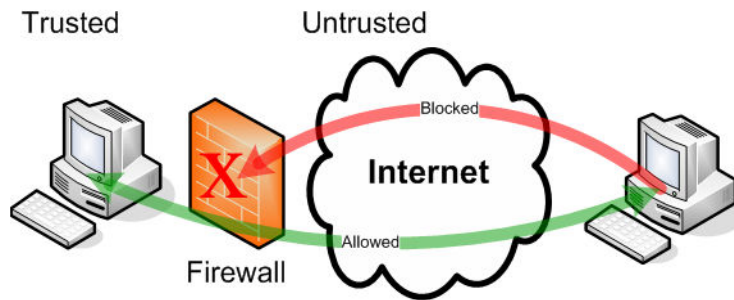
Mac Filter Rule List : filterSet1

1. permit any any vlanid 20
2. permit any any
3. permit any any ethertype ARP
4. deny any any

---

## Configuring a default firewall policy

The following figure shows an example of a default firewall policy.



**Figure 27: Default firewall example**

To configure a default firewall policy, perform the following steps.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To create the untrusted firewall zone, enter:

```
firewall internet
```

3. To add an interface to the untrusted zone, enter:

```
interface wan1  
exit
```

4. To create the trusted corp zone, enter:

```
firewall corp
```

5. To add an interface to the trusted zone, enter:

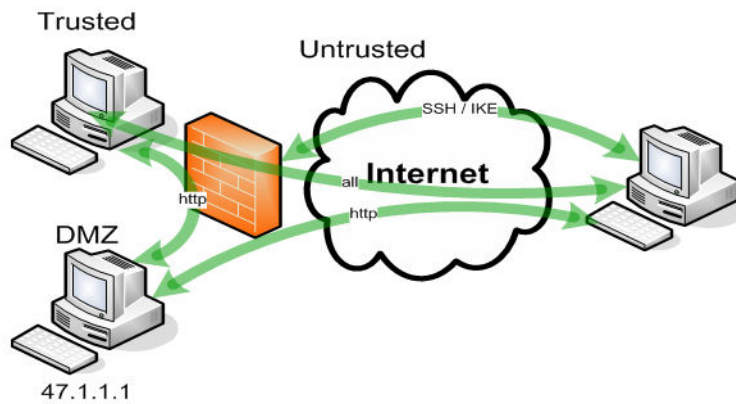
```
interface ethernet0/1
```

---

## Configuring a simple firewall policy with DMZ

The following figure shows a simple firewall policy with DMZ.





**Figure 28: Simple firewall policy with DMZ**

To configure the simple firewall policy shown, perform the following steps.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To select the IPsec proposal to configure, enter:

```
proposal <1-5> protocol [esp | ah]
```

5. To create the untrusted firewall zone, enter:

```
firewall internet
```

6. To add an interface to the untrusted zone, enter:

```
interface wan1
exit
```

7. To create the trusted corp zone, enter:

```
firewall corp
```

8. To add an interface to the trusted zone, enter:

```
interface ethernet0/1
exit
```

9. To create the trusted DMZ zone, enter:

```
firewall dmz
```

10. To add an interface to the DMZ zone, enter:

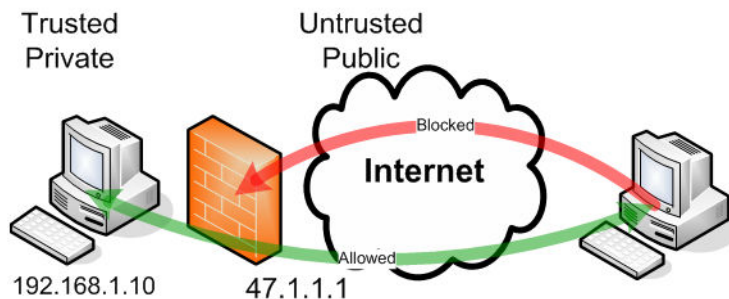
```
interface ethernet0/2
```

11. To define a policy for allowing HTTP traffic into a DMZ web server, enter:

```
policy 100 in service http address any any 47.1.1.1 32
```

## Configuring a simple PAT policy

The following figure shows a simple PAT policy.



**Figure 29: Simple PAT policy**

To configure the simple PAT policy shown, perform the following steps.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To select the IPsec proposal to configure, enter:

```
proposal <1-5> protocol [esp | ah]
```

5. To create the untrusted firewall zone, enter:

```
firewall internet
```

6. To add an interface to the untrusted zone, enter:

```
interface wan1  
exit
```

7. To create the trusted corp zone, enter:

```
firewall corp
```

8. To add an interface to the trusted zone, enter:

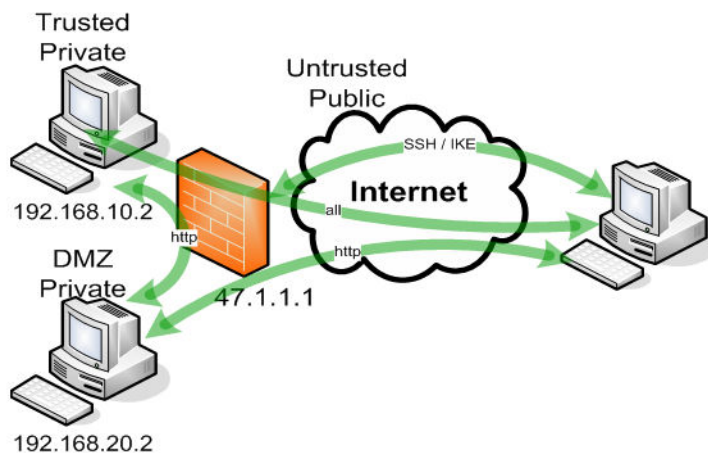
```
interface ethernet0/1
```

9. To create the outbound NAT IP policy, enter:

```
policy 1024 out nat-ip 47.1.1.1
```

## Configuring a PAT policy with an inbound forwarding policy

The following figure shows a PAT policy with an inbound forwarding policy.



**Figure 30: PAT policy with inbound forwarding policy**

To configure the PAT policy shown, perform the following steps.

### Procedure steps

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration for IPsec and IKE, enter:

```
crypto
```

3. To specify the IPsec policy to configure, enter:

```
ipsec policy <policy-name> <peer-address>
```

4. To select the IPsec proposal to configure, enter:

```
proposal <1-5> protocol [esp | ah]
```

5. To create the untrusted firewall zone, enter:

```
firewall internet
```

6. To add an interface to the untrusted zone, enter:

```
interface wan1
exit
```

7. To create the trusted corp zone, enter:

```
firewall corp
```

8. To add an interface to the trusted zone, enter:

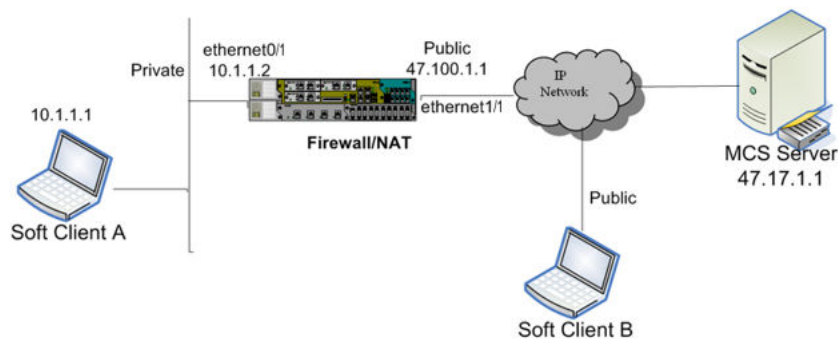
```
interface ethernet0/1
```

9. To create the outbound NAT IP policy, enter:

```
policy 1024 out nat-ip 47.1.1.1
```

## Configuring SIP ALG line-side

The following figure shows a SIP ALG line-side configuration.



**Figure 31: SIP ALG line-side example**

### Important:

Before you can configure SIP ALG line-side, you must enable SIP ALG globally .

To configure the firewall for the preceding configuration, perform the following steps.

### Prerequisites

- Enable SIP ALG globally.

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the firewall, enter:

```
firewall internet
interface ethernet1/1
```

```

firewall corp
interface ethernet0/1
policy 10 out nat-ip 47.100.1.1

```

## Configuring SIP ALG trunk-side

The following figure shows a SIP ALG trunk-side configuration.

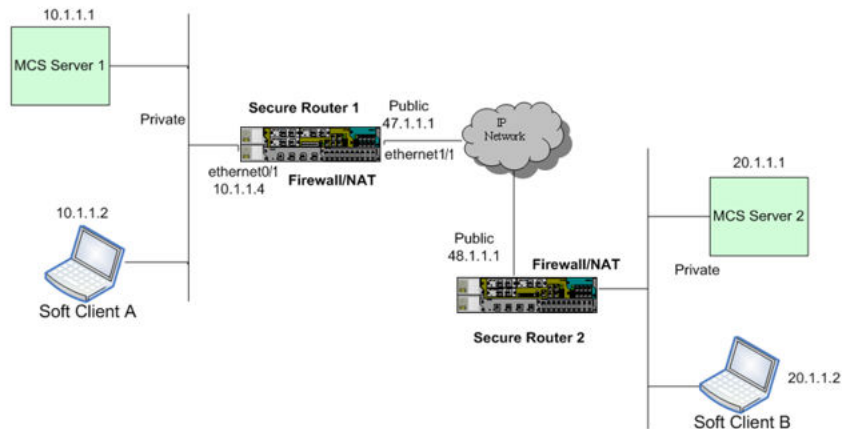


Figure 32: SIP ALG trunk-side

### Important:

Before you can configure SIP ALG trunk-side, you must enable SIP ALG globally .

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the firewall, enter:

```

firewall global
proxy-nat 10.1.1.1
firewall internet
interface ethernet1/1
firewall corp
interface ethernet0/1
policy 10 out nat-ip 47.1.1.100

```

(nat-ip different from ip address of public interface)

```

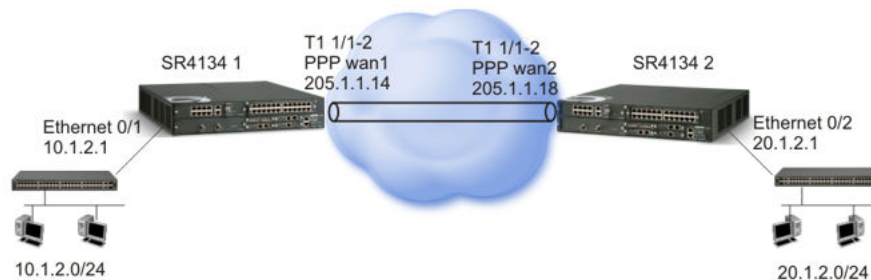
policy 20 in address 20.1.1.1 32 47.1.1.100 32 protocol
udp port 5060 any nat-ip 10.1.1.1
arp 47.1.1.100 00:50:52:8e:4a:01 published

```

(add arp for the nat-ip address)

## Configuring a Site-to-site IPsec VPN

The following figure shows a site-to-site VPN configuration.



**Figure 33: Site-to-site VPN example**

To configure the VPN shown in the preceding figure, perform the following steps.

### Configuring SR2330/4134 1

1. To configure the wan1 PPP bundle, enter:

```
interface bundle wan1
link t1 1/1-2
encapsulation ppp
ip address 205.1.1.14 30
crypto untrusted
exit
```

2. To configure the Ethernet port 0/1, enter:

```
interface ethernet 0/1
ip address 10.1.2.1 24
crypto trusted
exit
```

3. To configure the IKE policy, enter:

```
crypto
ike policy ike1 205.1.1.18
local-address 205.1.1.14
key certificatekey1
exit
```

4. To configure the IPsec policy, enter:

```
ipsec policy ipsec1 205.1.1.18
match address 10.1.2.0 24 20.1.2.0 24
enable
exit
```

5. To configure the internet firewall:

```
firewall internet
interface wan1
policy 100 in self service ike
policy 102 in self protocol icmp
```

**policy 102 in self protocol icmp** is optional. It can be helpful for debugging, but is not a necessary configuration.

6. To configure the corp firewall:

```
interface ethernet0/1
policy 101 address 20.1.2.0 24 any any
```

## Configuring SR2330/4134 2

1. To configure the wan2 PPP bundle, enter:

```
interface bundle wan2
link t1 1/1-2
encapsulation ppp
ip address 205.1.1.18 30
crypto untrusted
exit
```

2. To configure the Ethernet port 0/2, enter:

```
interface ethernet 0/2
ip address 20.1.2.1 24
crypto trusted
exit
```

3. To configure the IKE policy, enter:

```
crypto
ike policy ike1 205.1.1.14
local_address 205.1.1.18
key certificatekey1
pop
```

4. To configure the IPsec policy, enter:

```
crypto
ipsec policy ipsec1 205.1.1.14
match address 20.1.2.0 24 10.1.2.0 24
enable
exit
```

5. To configure the internet firewall:

```
firewall internet
interface wan2
policy 100 in self service ike
policy 102 in self protocol icmp
```

**policy 102 in self protocol icmp** is optional. It can be helpful for debugging, but is not a necessary configuration.

6. To configure the corp firewall:

```
interface ethernet0/2
policy 101 in address 10.1.2.0 24 any any
```

## Configuring a trust point for PKI

To configure a trust point for PKI, perform the following steps.

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To specify crypto configuration, enter:

```
crypto
```

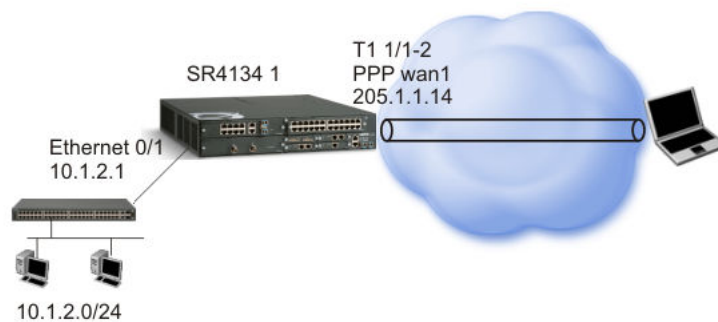
3. To configure the trustpoint, enter:

```
ca trustpoint sr4134
enrollment url http://certsrv.avaya.com/certsrv.dll
subject-name "cn=srsubName,o=avaya"
ip-address 192.168.118.33
fqdn sr4134.avaya.com
email sr4134@avaya.com
keypair srKey rsa 1024
crl query ldap://ldap.avaya.com/ldap
exit
exit
```

---

## Configuring a remote access IPsec VPN

The following figure shows a remote access IPsec VPN configuration.



**Figure 34: Remote access VPN example**

To configure the VPN shown in the preceding figure, perform the following steps.

1. To configure the wan1 PPP bundle, enter:

```
interface bundle wan1
link t1 1/1-2
encapsulation ppp
```



```
ip address 205.1.1.14 30
crypto untrusted
exit
```

2. To configure the Ethernet port 0/1, enter:

```
interface ethernet 0/1
ip address 10.1.2.1 24
crypto trusted
exit
```

3. To configure the dynamic IKE policy, enter:

```
crypto
dynamic
ike policy vpn_client modecfg-group
local-address 205.1.1.14
remote-id email-id "jim@domain.com"
key certificatekey1
```

4. To specify the address pool reserved for the remote clients, enter:

```
client configuration
address-pool 1 100.1.1.1 100.1.1.100
pop
```

5. To configure the dynamic IPsec policy, enter:

```
crypto
dynamic
ipsec policy to-vpn_client modecfg-group
match address 100.1.1.0 24
enable
exit
```

6. To configure the internet firewall:

```
firewall internet
interface wan1
policy 100 in self service ike
policy 101 in self protocol udp port any 4500
policy 200 in self protocol icmp
```

**policy 200 in self protocol icmp** is optional. It can be helpful for debugging, but is not a necessary configuration.

7. To configure the corp firewall:

```
interface ethernet0/1
policy 101 in address 100.1.1.1 100.1.1.100 any any
```

The source address specified here matches the address pool specified in step 5.

---

## Configuring a trust point for PKI

---

---

## Configuring a remote access VPN with L2TP server

---

To configure a remote access VPN with L2TP server, perform the following steps.

1. Configure the L2TP server:

```
interface l2tp-server l2tp1
ip address 192.169.100.1
remote-config my.domain.com
address-pool 192.168.100.2 192.168.100.100
dns 192.168.100.1
nbns 0.0.0.0
exit
```

2. Configure the users:

```
remote-user alice password1
remote-user bob password2
ipsec-protection l2tp1 192.168.10.2
exit l2tp-server
```

3. To configure the internet firewall:

```
firewall internet
interface wan1
policy 100 in self service ike
policy 101 in self protocol udp port any 4500
policy 102 in self protocol udp port any 1701
policy 200 in self protocol icmp
```

**policy 200 in self protocol icmp** is optional. It can be helpful for debugging, but is not a necessary configuration.

4. To configure the corp firewall:

```
firewall corp
interface ethernet0/1
policy 101 in address 192.168.100.2 192.168.100.100 any any
```

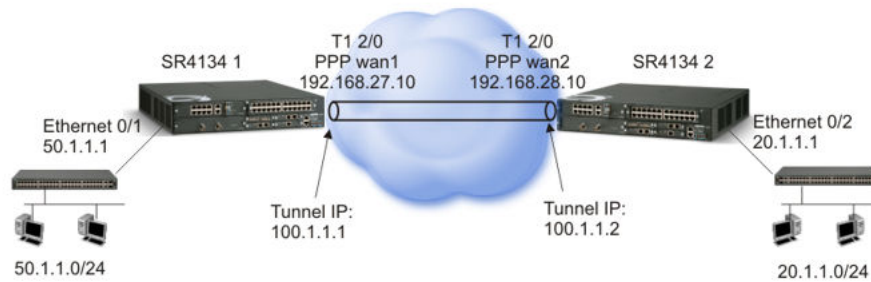
The source address specified here matches the address pool specified in step 2.

---

## Configuring an IPv4 tunnel

---

The following figure shows an IPv4 tunnel configuration.



**Figure 35: IPv4 tunnel example**

To configure the IPv4 tunnel shown in the preceding figure, perform the following steps.

### SR2330/4134 1

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the untrusted WAN interface, enter:

```
interface bundle wan1
link t1 2/
0 encapsulation ppp
ip address 192.168.27.10 255.255.255.0
crypto untrusted
exit
```

3. To configure the trusted Ethernet interface:

```
interface ethernet 0/1
ip address 50.1.1.1 255.255.255.0
crypto trusted
exit
```

4. To configure the tunnel interface, enter:

```
interface tunnel SanJose
ip address 100.1.1.1 24
tunnel source 192.168.27.10
tunnel destination 192.168.28.10
tunnel mode gre
crypto untrusted
```

5. To configure tunnel protection, enter:

```
tunnel protection toBilr Avaya
exit
```

6. To configure internet firewall, enter:

```
firewall internet
policy 100 in self protocol gre
policy 101 in self service ike
```

```
exit
exit
```

7. To configure corp firewall, enter:

```
firewall corp
policy 101 in
exit policy
exit firewall
```

## SR2330/4134 2

1. To enter the configuration mode, enter:

```
configure terminal
```

2. To configure the untrusted WAN interface, enter:

```
interface bundle wan1
link t1 2/0
encapsulation ppp
ip address 192.168.28.10 255.255.255.0
crypto untrusted
exit
```

3. To configure the trusted Ethernet interface:

```
interface ethernet 0/2
ip address 20.1.1.1 255.255.255.0
crypto trusted
exit
```

4. To configure the tunnel interface, enter:

```
interface tunnel Bilreca
ip address 100.1.1.2 24
tunnel source 192.168.28.10
tunnel destination 192.168.27.10
tunnel mode gre
crypto untrusted
```

5. To configure tunnel protection, enter:

```
tunnel protection toSanJ Avaya
exit
```

6. To configure internet firewall, enter:

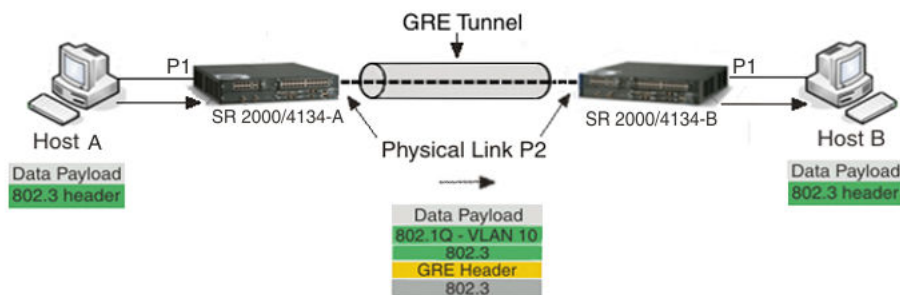
```
firewall internet
policy 100 in self protocol gre
policy 101 in self service ike
exit
exit
```

7. To configure corp firewall, enter:

```
firewall corp
policy 101 in
exit policy
exit firewall
```

## VLAN over a GRE tunnel configuration example

The following figure shows an example of a VLAN over a GRE tunnel configuration.



**Figure 36: VLAN over a GRE tunnel**

The following sections describe the steps required to configure a VLAN over a GRE tunnel, based on the topology shown in [Figure 36: VLAN over a GRE tunnel](#) on page 341.

### SR 2330/4134-A

1. To access configuration mode, enter:

```
configure terminal
```

2. To create VLAN 10, enter:

```
vlan database
vlan 10
exit
```

3. To configure port P1 for Host A, enter:

```
interface ethernet <0/1>
ip address 10.1.1.1 255.0.0.0
switchport
switchport mode access
switchport pvid 10
exit
```

4. To configure physical link P2, enter:

```
interface ethernet <0/2>
description "ethernetP2"
ip address 10.1.1.2 255.0.0.0
switchport
switchport mode access
switchport pvid 10
exit
```

5. To configure tunnel T1, enter:

```
interface tunnel T1
switchport
switchport mode trunk
switchport trunk allowed vlan 10
tunnel source 10.1.1.2
```

```
tunnel destination 20.1.1.2
exit
```

6. To assign an IP route for P2 to SR 2330/4134-B over tunnel T1, enter:

```
ip route 20.1.1.0/24 ethernetP2
exit
```

## SR 2330/4134-B

1. To access configuration mode, enter:

```
configure terminal
```

2. To create VLAN 10, enter:

```
vlan database
vlan 10
exit
```

3. To configure port P1 for Host B, enter:

```
interface ethernet <1/1>
ip address 20.1.1.1 255.0.0.0
switchport
switchport mode access
switchport pvid 10
exit
```

4. To configure physical link P2, enter:

```
interface ethernet <1/2>
description "ethernetP2"
ip address 20.1.1.2 255.0.0.0
switchport
switchport mode access
switchport pvid 10
exit
```

5. To configure tunnel T1, enter:

```
interface tunnel T1
switchport
switchport mode trunk
switchport trunk allowed vlan 10
tunnel source 20.1.1.2
tunnel destination 10.1.1.2
exit
```

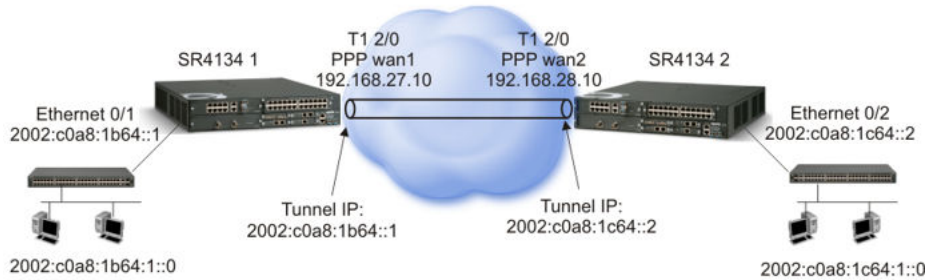
6. To assign an IP route for P2 to SR 2330/4134-A over tunnel T1, enter:

```
ip route 10.1.1.0/24 ethernetP2
exit
```

---

## Configuring an auto 6to4 tunnel

The following figure shows an auto 6to4 tunnel configuration.



**Figure 37: Auto 6to4 tunnel**

To configure an auto 6to4 tunnel, perform the following steps.

### SR2330/4134 1

1. To configure the IPv4-enabled interface, enter:

```
interface bundle wan1
link t1 2/0
encapsulation ppp
ip address 192.168.27.10 255.255.255.0
crypto untrusted
exit
```

2. To configure the IPv6-enabled interface, enter:

```
interface ethernet 0/1
ipv6 address 2002:c0a8:1b64::1/64
exit
```

3. To configure the tunnel interface, enter:

```
interface tunnel t1
ipv6 address 2002:c0a8:1b64::1/64
tunnel source 192.168.27.10
tunnel mode ipv6 6to4
exit
```

Tunnel destination is not required; packets are dynamically tunneled.

4. To add routes, enter:

```
ip route 192.168.28.0/24 wan1
ipv6 route 2002::/16 t1
```

### SR2330/4134 2

1. To configure the IPv4-enabled interface, enter:

```
interface bundle wan2
link t1 2/0
encapsulation ppp
ip address 192.168.28.10 255.255.255.0
```

```
crypto untrusted
exit
```

2. To configure the IPv6-enabled interface, enter:

```
interface ethernet 0/2
ipv6 address 2002:c0a8:1c64:1::2/64
exit
```

3. To configure the tunnel interface, enter:

```
interface tunnel t1
ipv6 address 2002:c0a8:1c64::1/64
tunnel source 192.168.28.10
tunnel mode ipv6 6to4
exit
```

Tunnel destination is not required; packets are dynamically tunneled.

4. To add routes, enter:

```
ip route 192.168.27.0/24 wan1
ipv6 route 2002::/16 t1
```

## Configuring the firewall for NAT and IPsec tunnels

This example shows how to properly configure the Secure Router 2330/4134 firewall to implement NAT as well as allow IPsec traffic within a Branch Office Tunnel created between two Secure Router 2330/4134s.

In the figure below, the link between the two networks is created through an IPsec Branch Office Tunnel. In addition, both networks are able to access the Internet through a NAT policy configured in each Secure Router 2330/4134. The NAT implementation is a Many to One PAT which allows multiple IP addresses to be mapped to one public address.

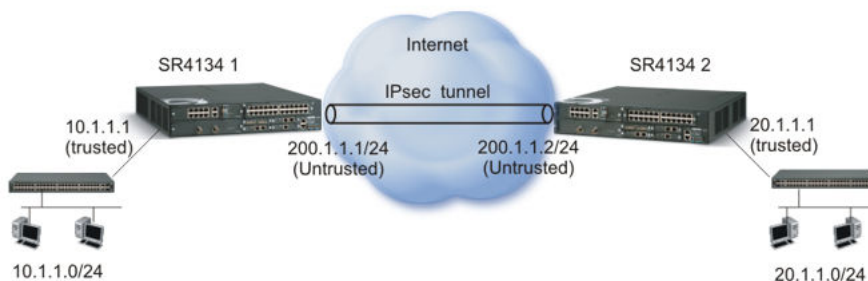


Figure 38: Firewall configuration for NAT and IPsec tunnel



**Firewall configuration for SR2000/4134 1**

1. To configure the internet firewall, enter:

```
configure terminal
firewall internet
policy 1000 in permit service ike self
exit
```

2. To add the untrusted WAN interface to the internet firewall, enter:

```
interface wan1
exit
```

3. To configure the corp firewall for incoming IPsec tunnel traffic, enter:

```
firewall corp
policy 1000 in permit address 20.1.1.0 24 10.1.1.0 24
exit
```

4. To configure the corp firewall for outgoing IPsec tunnel traffic, enter:

```
policy 1001 out permit address 10.1.1.0 24 20.1.1.0 24
exit
```

5. To configure the NAT for internet access, enter:

```
policy 1002 out permit address 10.1.1.2 10.1.1.254 any any nat-ip
200.1.1.1
exit
```

6. To add the trusted Ethernet interface to the corp firewall, enter:

```
interface ethernet0/1
exit
```

**Firewall configuration for SR2000/4134 2**

1. To configure the internet firewall, enter:

```
configure terminal
firewall internet
policy 1000 in permit service ike self
exit
```

2. To add the untrusted WAN interface to the internet firewall, enter:

```
interface wan1
exit
```

3. To configure the corp firewall for incoming IPsec tunnel traffic, enter:

```
firewall corp
policy 1000 in permit address 10.1.1.0 24 20.1.1.0 24
exit
```

4. To configure the corp firewall for outgoing IPsec tunnel traffic, enter:

```
policy 1001 out permit address 20.1.1.0 24 10.1.1.0 24
exit
```

5. To configure the NAT for internet access, enter:

```
policy 1002 out permit address 20.1.1.2 20.1.1.254 any
any nat-ip 200.1.1.2
exit
```

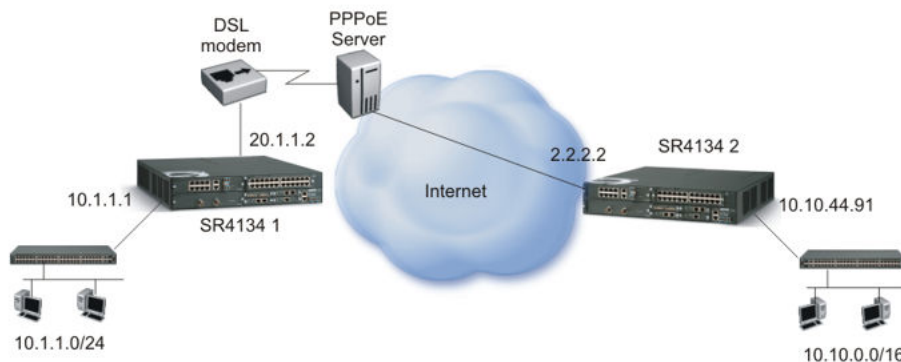
6. To add the trusted Ethernet interface to the corp firewall, enter:

```
interface ethernet0/1
exit
```

In this example, if the NAT rule is configured with a lower priority than the IPsec Tunnel traffic rules, all packets with source address within the configured range first match the NAT rule. Additionally, if no rule is configured to match the outgoing traffic sent through the IPsec Tunnel, then the outgoing packets with source address within the configured range always match the NAT rule, even if it is configured with a higher priority than the incoming IPsec Tunnel traffic rule. In either situation, no packets would ever be directed to the IPsec Tunnel.

The configuration above avoids these scenarios from occurring.

## Configuring a PPPoE client



**Figure 39: PPPoE client configuration**

### SR2330/4134 1

To configure a PPPoE client on SR2330/4134 1, perform the following steps.

1. To configure the internet interface, enter:

```
configure terminal
interface ethernet 0/2
ip address 20.1.1.2 255.255.255.0
crypto untrusted
exit
```

2. To configure the trusted interface, enter:

```
interface ethernet 0/1
ip address 10.1.1.1 255.255.255.0
```

```
crypto trusted
exit
```

3. To configure the PPPoE client, enter:

```
interface virtual-access test
ip negotiated
protocol pppoe
pppoe ethernet 0/2
ppp authentication pap sent-username test password test
exit
```

4. To add routes, enter:

```
ip route 0.0.0.0 0.0.0.0 test 10
```

5. To configure IKE and IPsec policies for IPsec over PPPoE, enter:

```
crypto
ike policy test 2.2.2.2
local-id domain-name client.Avaya.com
local-address 0.0.0.0
key testing12345
exit policy
ipsec policy test 2.2.2.2
match address 10.10.0.0 255.255.0.0 10.1.1.0 255.255.255.0
exit policy
exit crypto
```

## SR2330/4134 2

To configure the peer VPN gateway on SR2330/4134 2, perform the following steps.

1. To configure the internet interface, enter:

```
configure terminal
interface bundle wan1
link t1 1/1
encapsulation ppp
ip address 2.2.2.2 255.255.255.0
crypto untrusted
exit
```

2. To configure the trusted interface, enter:

```
interface ethernet 0/1
ip address 10.10.44.91 255.255.255.0
crypto trusted
exit
```

3. To add routes, enter:

```
ip route 0.0.0.0 0.0.0.0 2.2.2.1 1
```

4. To configure IKE and IPsec policies, enter:

```
crypto
ike policy test 0.0.0.0
remote-id domain-name client.Avaya.com
local-address 2.2.2.2
key testing12345
exit policy
ipsec policy test 0.0.0.0
match address 10.10.0.0 255.255.0.0 10.1.1.0 255.255.255.0
exit policy
exit crypto
```

## Secure Router 2330/4134 configuration for dynamic route exchange over IPsec tunnel interoperability with VPN Router

Both Secure Router and VPN router currently support dynamic routing over IPsec. Secure router configuration for dynamic route exchange over IPsec Tunnel allows interoperability by using IP-on-IP over a transport mode IPsec connection.

### Capabilities

- IPsec transport mode is used, not tunnel mode
- For OSPF configurations, the Secure Router 2330/4134 default IPIP tunnel MTU needs to be set to 1500 to match the VPN Router tunnel MTU.
- If both “ip mtu” and “tunnel path-mtu-discovery” are configured and enabled on the Secure Router 2330/4134 the MTU value set by “ip mtu” configuration takes effect.

### Secure Router configuration for BGP

Configure the Secure Router 2330/4134 for BGP as follows:

1. To configure an Ethernet interface for source traffic, enter:

```
interface ethernet 6/12
ip address 10.10.10.1 24
crypto trusted
exit
```

2. To configure an Ethernet interface as a tunnel source interface, enter:

```
interface ethernet 0/1
ip address 192.168.26.100 24
crypto untrusted
exit
```

3. To configure the tunnel interface, enter:

```
interface tunnel toCes
tunnel mode ipip
ip address 100.1.1.1 24
tunnel source 192.168.26.100
tunnel destination 192.168.27.100
tunnel protection toCes Avaya
crypto untrusted
exit
```

4. To redistribute the tunnel route into OSPF, enter:

```
router ospf
redistribute connected
exit
```

5. To configure a loopback interface, enter:

```
interface loopback 111
ip address 50.1.1.1 24
exit
```

6. To configure BGP, enter:

```
router bgp 100
neighbor 50.1.1.10 update-source 50.1.1.1
exit
exit
```

7. To configure the IP route, enter:

```
ip route 50.1.1.10/32 toCes
```

## Secure Router configuration for OSPF

Configure the Secure Router 2330/4134 for OSPF as follows:

1. To configure an Ethernet interface for source traffic, enter:

```
interface ethernet 6/12
ip address 10.10.10.1 24
crypto trusted
exit
```

2. To configure an Ethernet interface as a tunnel source interface, enter:

```
interface ethernet 0/1
ip address 192.168.26.100 24
crypto untrusted
exit
```

3. To configure the tunnel interface, enter:

```
interface tunnel toCes
tunnel mode ipip
ip address 100.1.1.1 24
ip mtu 1500
tunnel source 192.168.26.100
tunnel destination 192.168.27.100
tunnel protection toCes Avaya
crypto untrusted
exit
```

4. To configure OSPF, enter:

```
router ospf 1
network toCes
area 0
exit
```

## Secure Router configuration for RIPv2

Configure the Secure Router 2330/4134 for RIPv2 as follows:

1. To configure an Ethernet interface for source traffic, enter:

```
interface ethernet 6/12
ip address 10.10.10.1 24
crypto trusted
exit
```

2. To configure an Ethernet interface as a tunnel source interface, enter:

```
interface ethernet 0/1
ip address 192.168.26.100 24
```

```
crypto untrusted
exit
```

3. To configure the tunnel interface, enter:

```
interface tunnel toCes
tunnel mode ipip
ip address 100.1.1.1 24
tunnel source 192.168.26.100
tunnel destination 192.168.27.100
tunnel protection toCes Avaya
crypto untrusted
exit
```

4. To configure a loopback interface, enter:

```
interface loopback 111
ip address 192.168.26.100 32
exit
```

5. To configure the router ID, enter:

```
router-id 192.168.26.100
```

6. To configure RIP routes, enter:

```
ip route 192.168.27.0/24 192.168.26.101
router rip
network interface toCes
mode 2
exit
```

7. To configure the corp firewall, enter:

```
firewall corp
policy 101 in
exit
exit
```

8. To configure internet firewall, enter:

```
firewall internet
policy 100 in self
exit
exit
```

---

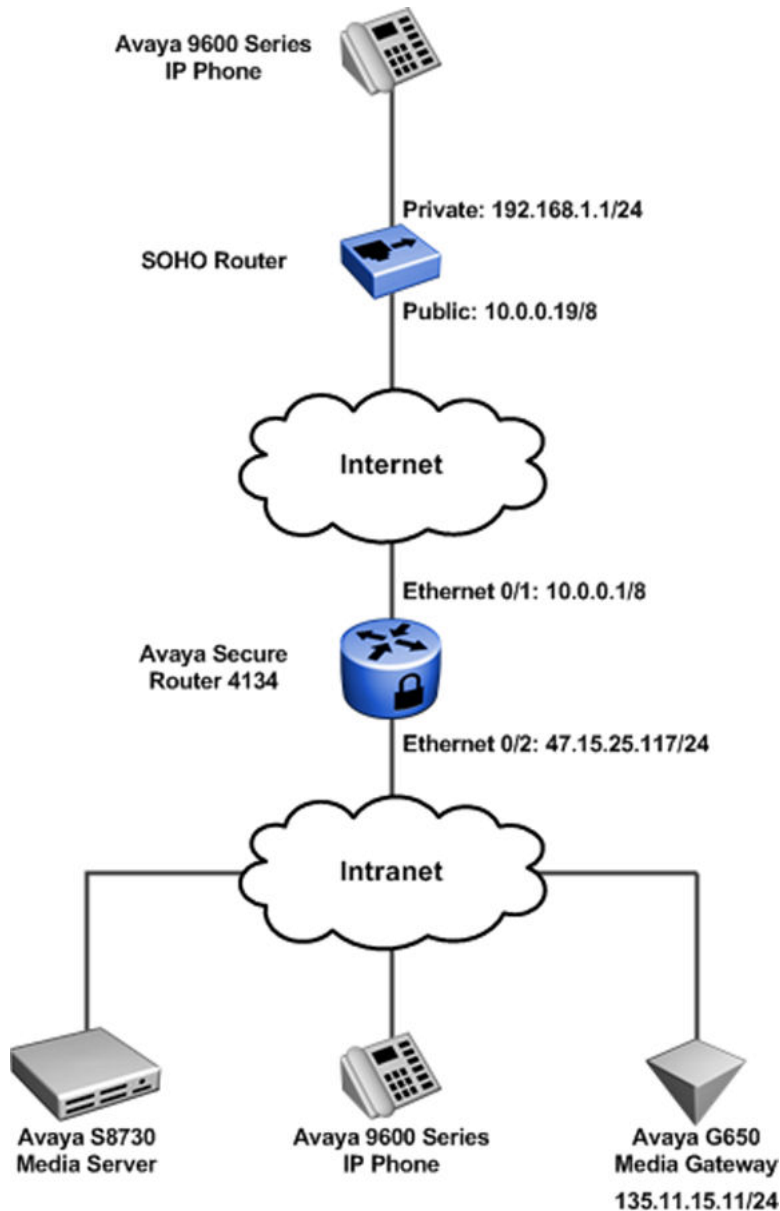
## IP phone configuration for Secure Router 2330/4134

The following sections provide configuration examples for the Avaya 9600 Series IP phone over IPsec tunnel to the Secure Router 2330/4134.

---

## Secure Router 2330/4134 Interface, IPsec and Firewall configuration examples

The following network topology was used to validate the configuration outlined in this section:



**Figure 40: Network topology example**

The following sections highlight the Interface, IPsec and Firewall configuration defined on the Secure Router 4134 to support IPsec VPN tunnels from Avaya 9600 Series IP Phones.

### Interface configuration

```
interface ethernet 0/1
  ip address 10.0.0.1 255.0.0.0
  ip rip send version 2
  ip rip receive version 2
  aaa
  exit aaa
## Public Interface
crypto untrusted
qos
```

```

        chassis
        exit chassis
    exit qos
exit ethernet
interface ethernet 0/2
    ip address 47.17.25.117 255.255.255.0
## proxy arp required so SR arps on behalf of IPsec RAS clients
ip proxy_arp
    aaa
    exit aaa
## Private Interface
crypto trusted
qos
    chassis
    exit chassis
    exit qos
exit Ethernet

```

## IPsec configuration

```

crypto
    dynamic
        exit dynamic
## contivity-iras used for interoperability with 9600
contivity-iras
    ike policy cont1
## local-address is the public interface (WAN) of the Secure Router
    local-address 10.0.0.1
## remote-id username test strings must be quoted, followed by password
    remote-id user-name "voip" voip
    proposal 1
        dh-group group2
        encryption-algorithm 3des-cbc
    exit proposal
    client configuration
        address-pool 1 47.17.25.120 47.17.25.125
## private-side address is the LAN interface of the router. Crypto TRUSTED.
        private-side-address 47.17.25.117
        keepalive
            exit keepalive
        split-tunnel
            mode enabled
            network 47.17.25.0 24
            exit split-tunnel
        nat-keepalive 40
        exit configuration
    exit policy
    ipsec policy cont1
        proposal 1
            lifetime seconds 3600
            exit proposal
        exit policy
    exit contivity-iras
no keepalive mode periodic
pmtu
    exit pmtu
qos
    chassis
    exit chassis
    exit qos
exit crypto

```



## Firewall configuration

```

firewall internet
  interface ethernet0/1
    policy 100 in permit service ike self
    exit policy
    policy 101 in permit protocol udp port 4500 4500 self
    exit policy
## Allow encapsulated packets for ipsec policy processing.
  policy 102 in permit address 47.17.25.120 47.17.25.125 47.17.25.117 32 self
  exit policy
## Permit USDP for ESP traffic (IPsec)
  policy 103 in permit protocol tcp port any 17 self
  exit policy
## Added for ping testing during setup, an be left out, but SR will not reply to ping.
  policy 104 in permit protocol icmp self
  exit policy
exit firewall
firewall corp
  interface ethernet0/2 ethernet6/2
  policy 10 in permit
  exit policy
## Allow encapsulated packets for ipsec policy processing.
  policy 100 in permit address 47.17.25.120 47.17.25.125 47.17.25.0 24
  exit policy
  policy 1024 out permit
  exit policy
exit firewall

```

## Avaya 9600 Series IP Phone network settings

The following table highlights the network settings defined on the Avaya 9600 Series IP Phone to establish a secure IPsec VPN tunnel to the Secure Router 4134:

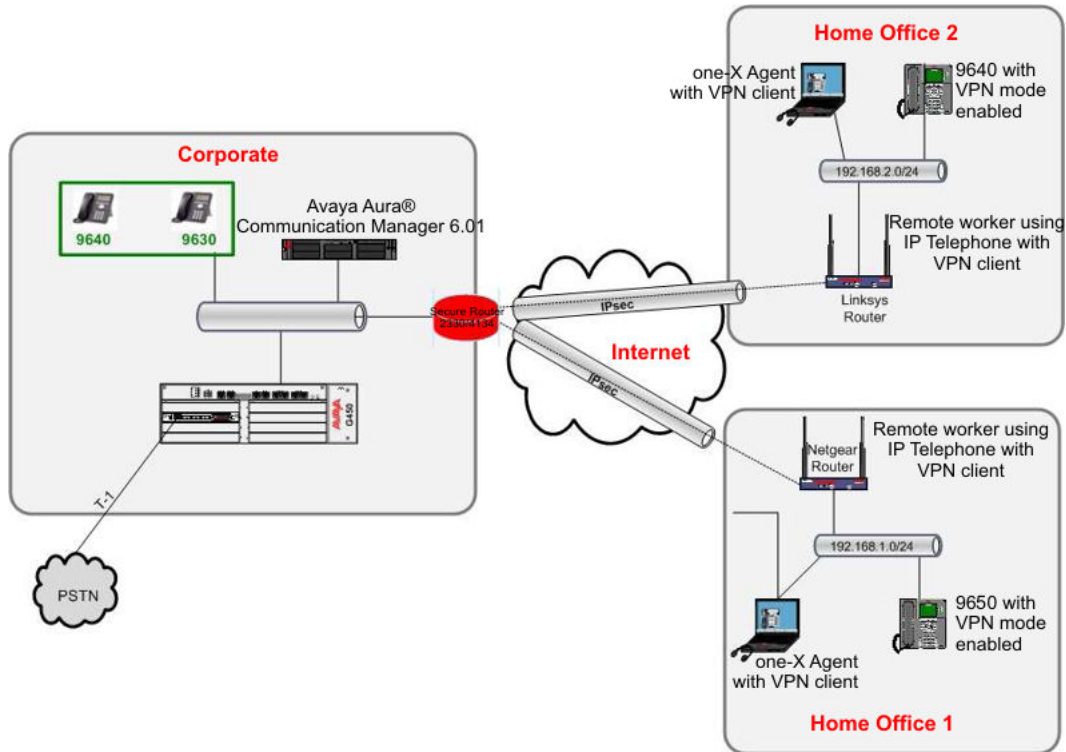
| Prompt                    | Value             |
|---------------------------|-------------------|
| VPN                       | Enabled           |
| VPN Vendor                | Nortel            |
| Gateway Address           | 10.0.0.1          |
| External Phone IP Address | 0.0.0.0           |
| External Router           | 192.168.1.1       |
| External Subnet Mask      | 255.255.255.0     |
| Encapsulation             | Disabled          |
| Auth Type                 | Local Credentials |
| VPN User Type             | Any               |
| VPN User                  | voip              |
| Password Type             | Save in Flash     |
| User Password             | voip              |

| Prompt                         | Value      |
|--------------------------------|------------|
| IKE ID Type                    | KEY_ID     |
| IKE Xchg Mode                  | Aggressive |
| IKE DH Group                   | 2          |
| IKS Auth Algorithm             | 3DES       |
| IKE Encryption Algorithm       | SHA-1      |
| IKE Config Mode                | Enabled    |
| IPsec PFS DH Group             | NO PFS     |
| IPsec Encryption Algorithm     | 3DES       |
| IPsec Authentication Algorithm | SHA-1      |
| Protected n/w                  | 0.0.0.0/0  |
| IKE over TCP                   | Never      |

---

## Configuring an IPSec Tunnel between Avaya 96xx Series IP Phones and the Avaya Secure Router 2330/4134

The following network topology was used to validate the configuration outlined in this section:



**Figure 41: Avaya Secure Router 4134 as a VPN Gateway for Home Office Users**

To implement IPSec VPN on the SR4134, perform the following configuration tasks:

- Assign host name, configure Ethernet ports and default route
- Configure default routing
- Configure Untrusted and Trusted firewall
- Create IKE policies
  - Configure remote-id
  - Configure proposal 1
  - Configure client configuration

### Assigning host name, Configuring Ethernet ports and Default Route

Change the hostname to sr4134-1 (sr2330-1 for the SR 2330). Configure trusted and untrusted Ethernet interfaces. Configure the default route to go out the untrusted interface.

```
hostname sr4134-1

interface ethernet 0/1
  description trusted
  ip address 10.80.70.254 255.255.255.0
  ip proxy-arp
  crypto trusted
  exit Ethernet

interface ethernet 0/2
  description untrusted
  ip address 192.45.130.1 255.255.255.0
```

```
crypto untrusted
exit ethernet
```

## Configuring Untrusted (Internet) firewall

This example is a minimal firewall configuration.

```
firewall internet
  interface ethernet0/2
  policy 110 in permit service ike self
    exit policy
  policy 115 in permit protocol udp port any 4500 self
    exit policy
  policy 117 in permit address 10.80.70.230 10.80.70.239 any any self
    exit policy
  policy 120 in permit address 10.80.70.240 10.80.70.250 any any self
    exit policy
  policy 130 in permit protocol tcp port any 17 self
    exit policy
  policy 140 in permit protocol icmp self
    exit policy
exit firewall
```

## Configuring Trusted (Corp) firewall

```
firewall corp
  interface ethernet0/1
  policy 100 in permit
    exit policy
  policy 107 out permit address 10.80.70.230 10.80.70.239 any any
    exit policy
  policy 108 in permit address 10.80.70.230 10.80.70.239 any any
    exit policy
  policy 109 out permit address 10.80.70.240 10.80.70.250 any any
    exit policy
  policy 110 in permit address 10.80.70.240 10.80.70.250 any any
    exit policy
  policy 1024 out permit
    exit policy
exit firewall
```

## Creating IKE Policies

Two IKE policies were configured. The ip9600 policy is for the 96xx series IP phones running the VPN firmware. The vpnclient policy is used by the Windows VPN client. The **ipsec policy ip9600** and **ipsec policy vpnclient** are created as a result of the IKE policies.

```
crypto
  dynamic
    exit dynamic
  contivity-iras
    ike policy ip9600
      local-address 192.45.130.1
      remote-id user-name "ladgjm" ladgjm
      proposal 1
        dh-group group2
        encryption-algorithm 3des-cbc
        exit proposal
      client configuration
        address-pool 1 10.80.70.240 10.80.70.250
        private-side-address 10.80.70.254
        keepalive
          enable
          interval 60
        exit keepalive
```

```

        split-tunnel
            mode enabled
            network 10.80.70.0 24
        exit split-tunnel
        nat-keepalive 20
        exit configuration
    exit policy

    ike policy vpnclient
        local-address 192.45.130.1
        remote-id user-name "client01" client123
        remote-id user-name "client02" client123

    proposal 1
        dh-group group2
        encryption-algorithm 3des-cbc
        exit proposal
    client configuration
        address-pool 1 10.80.70.230 10.80.70.239
        private-side-address 10.80.70.254
        keepalive
            enable
            interval 60
        exit keepalive
        split-tunnel
            mode enabled
            network 10.80.70.0 24
        exit split-tunnel
        nat-keepalive 20
        exit configuration
    exit policy
ipsec policy ip9600
    proposal 1
        lifetime seconds 3600
    exit proposal
    exit policy
ipsec policy vpnclient
    proposal 1
        lifetime seconds 3600
    exit proposal
    exit policy
exit contivity-iras

```



# Chapter 20: Default settings

The following sections list the command defaults for the Avaya Secure Router 2330/4134 security features.

---

## Firewall command defaults

The following table lists the default settings for firewall commands.

**Table 185: Firewall command defaults**

| Commands              | Defaults |
|-----------------------|----------|
| configure             |          |
| -- firewall global    | N/A      |
| -- interface          | N/A      |
| -- object             | N/A      |
| -- address            | none     |
| -- nat-pool           | none     |
| -- ftp-filter         | none     |
| -- http-filter        | none     |
| -- smtp-filter        | none     |
| -- rpc-filter         | none     |
| -- service            | none     |
| -- schedule           | none     |
| -- port-trigger       | none     |
| -- stealth-mode       | disabled |
| -- reset-invalid-acks | enabled  |
| -- dos-protect        | N/A      |
| -- syn-flooding       | enabled  |
| -- source-routing     | disabled |
| -- win-nuke  --       | disabled |

| Commands                  | Defaults |
|---------------------------|----------|
| -- ftp-bounce             | disabled |
| -- icmp-error             | enabled  |
| -- dns-replay-attack      | disabled |
| -- ip-unaligned-timestamp | disabled |
| -- tcp-seq-number-predict | disabled |
| -- tcp-seq-number-range   | disabled |
| -- mime-flood             | disabled |
| -- enable-all             | N/A      |
| -- algs                   |          |
| -- aim                    | disabled |
| -- aimudp                 | disabled |
| -- ftp                    | disabled |
| -- h323                   | disabled |
| -- msn                    | disabled |
| -- pptp                   | disabled |
| -- rpc                    | disabled |
| -- rtsp554                | disabled |
| -- rtsp7070               | disabled |
| -- sip                    | disabled |
| -- sip-tcp                | disabled |
| -- smtp                   | disabled |
| -- tftp                   | disabled |
| -- web                    | disabled |
| -- cuseeme                | disabled |
| -- dns                    | N/A      |
| -- enable                 | disabled |
| -- pool                   | none     |
| -- gatekeeper             | disabled |
| -- ike                    | disabled |
| -- ils                    | disabled |
| -- ils2                   | disabled |



| Commands                | Defaults                                                                                  |
|-------------------------|-------------------------------------------------------------------------------------------|
| -- irc                  | disabled                                                                                  |
| -- msgtcp               | disabled                                                                                  |
| -- msgudp               | disabled                                                                                  |
| -- mszone               | disabled                                                                                  |
| -- n2p                  | disabled                                                                                  |
| -- n2pe                 | disabled                                                                                  |
| -- nntp                 | disabled                                                                                  |
| -- pcanywhere           | disabled                                                                                  |
| -- sql                  | disabled                                                                                  |
| -- enable-all           | N/A                                                                                       |
| -- enable-typical       | N/A                                                                                       |
| -- timeout              | note: tcp = 600, udp = 60, icmp=60, tcp-reset=20, ftp-inactivity=600, dns-inactivity=600. |
| -- general              | none                                                                                      |
| -- service              | none                                                                                      |
| -- url-key-filter       | N/A                                                                                       |
| -- max-connection-limit | note: self=2048, internet=7500, corp=2500                                                 |
| -- ip-reassembly        | N/A                                                                                       |
| -- enable               | enabled                                                                                   |
| -- packet-size          | 65535                                                                                     |
| -- fragment-size        | 28                                                                                        |
| -- fragment-count       | 44                                                                                        |
| -- timeout              | 60                                                                                        |
| -- logging              |                                                                                           |
| -- vpn                  | 100                                                                                       |
| -- attacks              | 100                                                                                       |
| -- policy               | 1                                                                                         |
| -- bypass-trusted       | disabled                                                                                  |
| -- hairpinning-Selfip   | disabled                                                                                  |
| -- policy               |                                                                                           |

| Commands                  | Defaults |
|---------------------------|----------|
| -- max-connection-limit   | none     |
| -- connection-rate        | none     |
| -- apply-object           | none     |
| -- policing               | none     |
| -- bandwidth              | none     |
| -- enable                 | enabled  |
| -- connection-reservation | none     |
| -- nat-failover           | none     |

---

## Packet filter defaults

By default, no packet filters are enabled nor are there are any default settings configured for any packet filter commands.

---

## IPSec VPN default settings

The following table lists the default settings for IPSec VPN commands.

**Table 186: IPSec VPN command defaults**

| Commands           | Default settings     |
|--------------------|----------------------|
| -- configure       |                      |
| -- crypto          |                      |
| -- ike             | None                 |
| -- policy          | None                 |
| -- local-id        | ipv4 address (local) |
| -- remote-id       | ipv4 address (peer)  |
| -- local-address   | None                 |
| -- initial-contact | ON                   |
| -- key             | None                 |
| -- mode            | Main                 |

| Commands                 | Default settings                       |
|--------------------------|----------------------------------------|
| -- pfs                   | off                                    |
| -- ocsp                  | off                                    |
| -- exchange-type         | both initiator-responder               |
| -- proposal              | 1                                      |
| -- hash-algorithm        | sha1                                   |
| -- authentication-method | pre-shared-key                         |
| -- dh-group              | 2                                      |
| -- encryption-algorithm  | 3des-cbc                               |
| -- lifetime              |                                        |
| -- seconds               | 86400                                  |
| -- kilobytes             | unlimited                              |
| -- ipsec                 | None                                   |
| -- policy                | None                                   |
| -- match                 | None                                   |
| -- address               | None                                   |
| -- enable                | on                                     |
| -- pfs-group             | off                                    |
| -- proposal              | 1, esp                                 |
| -- hash-algorithm        | sha1-hmac                              |
| -- encryption-algorithm  | 3des-cbc                               |
| -- mode                  | tunnel                                 |
| -- lifetime              |                                        |
| -- seconds               | 3600                                   |
| -- kilobytes             | 4194303                                |
| -- anti-replay           | enabled                                |
| -- dynamic               | None                                   |
| -- ike                   | None                                   |
| -- policy                | None for name, modecfg-group for group |
| -- local-address         | None                                   |
| -- remote-id             | None                                   |
| -- local-id              | ipv4 address (local)                   |



| Commands                     | Default settings |
|------------------------------|------------------|
| -- kilobytes                 | 4194303          |
| -- keepalive                 |                  |
| -- enable                    | On               |
| -- transmit-interval         | 30               |
| -- retry-interval            | 10               |
| -- ca                        | N/A              |
| -- trustpoint                | N/A              |
| -- enrollment                | N/A              |
| -- terminal                  | None             |
| -- url                       | None             |
| -- subject-name              | None             |
| -- password                  | None             |
| -- ip-address                | None             |
| -- fqdn                      | None             |
| -- email                     | None             |
| -- keypair                   | None             |
| -- ocsp                      | None             |
| -- url                       | None             |
| -- nonce                     | None             |
| -- signature                 | None             |
| -- crl                       | None             |
| -- query                     | None             |
| -- authenticate              | None             |
| -- enroll                    | None             |
| -- import                    | None             |
| -- router-certificate        | None             |
| -- responder-certificate     | None             |
| -- crl                       | None             |
| -- request                   | None             |
| -- pmtu                      | N/A              |
| -- unsecured-icmp-processing | disabled         |

| Commands         | Default settings                                                     |
|------------------|----------------------------------------------------------------------|
| -- threshold-mtu | 576                                                                  |
| -- df-bit        | no default value - but default behavior is 'copy' on all interfaces. |

The following table lists the default settings for L2TP server commands.

**Table 187: L2TP server command defaults**

| Commands                 | Defaults                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -- configure             |                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| - -interface l2tp-server |                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| -- ip                    | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| -- address               | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -- remote-config         | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -- address-pool          | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| - dns-                   | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -- nbns                  | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -- remote-user           | none                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| -- shutdown              | no shutdown = 'up'                                                                                                                                                                                                                                                                                                                                                                                                                           |
| -- ipsec-protection      | note: if you do not provide a key on this command, then you get an ike policy of AggressiveMode/RSA-Sig/DHGroup2/3DES/SHA with remote-id of der-encoded-dn="" and local-id = ipaddress-of-the-physical-int. On the other hand, if you enter a key on this line, then you get an ike policy of AggressiveMode/PSK/DHGroup2/3DES/SHA with remote-id of ipaddress-type=0.0.0.0. In either case, you get an ipsec policy of ESP-Tunnel/3des/sha. |

---

## PPPoE default settings

The following table lists the defaults settings for PPPoE commands.

**Table 188: PPPoE command defaults**

| Commands                  | Defaults           |
|---------------------------|--------------------|
| -- configure              |                    |
| -interface virtual-access |                    |
| -- ip                     | N/A                |
| -- address                | none               |
| -- negotiated             | none               |
| -- protocol               | none               |
| -- pppoe                  | N/A                |
| -- pppoe                  | N/A                |
| -- ethernet               | none               |
| -- ac-name                | none               |
| -- ppp                    | N/A                |
| -- authentication         | none               |
| -- keepalive              | 10                 |
| -- shutdown               | no shutdown = 'up' |

---

## GRE and IPIP tunnel default settings

The following table lists the default settings for GRE and IPIP tunnel commands.

**Table 189: GRE and IPIP tunnel command defaults**

| Commands             | Defaults                                                                                                                                                                                     |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -- configure         |                                                                                                                                                                                              |
| -- interface tunnel  |                                                                                                                                                                                              |
| -- tunnel protection | none Note: If you give the tunnel protection command, you will wind up with an ike policy of MainMode/PreSharedKeyAuth/DH-group2/3DES/SHA and an ipsec policy of ESP-TransportMode/3DES/SHA/ |
| -- crypto            | none                                                                                                                                                                                         |

