JUNIPER NETWORKS

Application-Level Gateways for JSF

Published: 2011-02-06
Part Number: , Revision

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Application-Level Gateways for JSF

Revision History
16 December 2010—Application-Level Gateways for JSF

The information in this document is current as of the date listed in the revision history.

END USER LICENSE AGREEMENT

READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. **The Parties.** The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer's principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer's principal office is located outside the Americas) (such applicable entity being referred to herein as "Juniper"), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software ("Customer") (collectively, the "Parties").

2. **The Software.** In this Agreement, "Software" means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. "Software" also includes updates, upgrades and new releases of such software. "Embedded Software" means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. **License Grant.** Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.

b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.

c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.

d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.

e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the

Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. **Audit.** Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. **Ownership.** Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement. TO THE MAXIMUM EXTENT PERMITTED BY LAW, JUNIPER SHALL NOT BE LIABLE FOR ANY LOST PROFITS, LOSS OF DATA, OR COSTS OR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, THE SOFTWARE, OR ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. IN NO EVENT SHALL JUNIPER BE LIABLE FOR DAMAGES ARISING FROM UNAUTHORIZED OR IMPROPER USE OF ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. EXCEPT AS EXPRESSLY PROVIDED IN THE WARRANTY STATEMENT TO THE EXTENT PERMITTED BY LAW, JUNIPER DISCLAIMS ANY AND ALL WARRANTIES IN AND TO THE SOFTWARE (WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE), INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT DOES JUNIPER WARRANT THAT THE SOFTWARE, OR ANY EQUIPMENT OR NETWORK RUNNING THE SOFTWARE, WILL OPERATE WITHOUT ERROR OR INTERRUPTION, OR WILL BE FREE OF VULNERABILITY TO INTRUSION OR ATTACK. In no event shall Juniper's or its suppliers' or licensors' liability to Customer, whether in contract, tort (including negligence), breach of warranty, or otherwise, exceed the price paid by Customer for the Software that gave rise to the claim, or if the Software is embedded in another Juniper product, the price paid by Customer for such other product. Customer acknowledges and agrees that Juniper has set its prices and entered into this Agreement in reliance upon the disclaimers of warranty and the limitations of liability set forth herein, that the same reflect an allocation of risk between the Parties (including the risk that a contract remedy may fail of its essential purpose and cause consequential loss), and that the same form an essential basis of the bargain between the Parties.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer's payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder. Customer shall comply with all applicable tax laws and regulations, and Customer will promptly pay or reimburse Juniper for all costs and damages related to any liability incurred by Juniper as a result of Customer's non-compliance or delay with its responsibilities herein. Customer's obligations under this Section shall survive termination or expiration of this Agreement.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. **Commercial Computer Software.** The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14(ALT III) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL")), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at http://www.gnu.org/licenses/gpl.html, and a copy of the LGPL at http://www.gnu.org/licenses/lgpl.html .

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentés confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattaché, soient redigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Table of Contents

List of Tables

PART 1

Overview

CHAPTER 1

Application-Level Gateways

Application-Level Gateways for JSF

An *Application Layer Gateway (ALG)* is a software component that is designed to manage specific protocols such as FTP on Juniper Networks devices running Junos OS. The ALG module is responsible for Application-Layer aware packet processing.

ALG functionality can be triggered either by a service or application configured in the security policy:

- A *service* is an object that identifies an application protocol using Layer 4 information (such as standard and accepted TCP and UDP port numbers) for an application service (such as Telnet, FTP, SMTP, and HTTP).
- An *application* specifies the Layer 7 application that maps to a Layer 4 service.

A predefined service already has a mapping to a Layer 7 application. However, for custom services, you must link the service to an application explicitly, especially if you want the policy to apply an ALG.

ALGs for packets destined to well-known ports are triggered by service type. The ALG intercepts and analyzes the specified traffic, allocates resources, and defines dynamic policies to permit the traffic to pass securely through the device:

1. When a packet arrives at the device, the flow module forwards the packet according to the security rule set in the policy.
2. If a policy is found to permit the packet, the associated service type or application type is assigned and a session is created for this type of traffic.
3. If a session is found for the packet, no policy rule match is needed. The ALG module is triggered if that particular service or application type requires the supported ALG processing.

The ALG also inspects the packet for embedded IP address and port information in the packet payload, and performs Network Address Translation (NAT) processing if necessary. The ALG also opens a gate for the IP address and port number to permit data exchange for the session. The control session and data session can be coupled to have the same timeout value, or they can be independent.

ALGs are supported on chassis clusters. For information about chassis clusters, see Chassis Cluster Overview.

Related Documentation

- *Junos OS Feature Support Reference for SRX Series and J Series Devices*
- Understanding ALG Types
- Understanding RPC ALGs

ALG Descriptions

This section includes details about the ALGs. It includes the following:

Basic TCP ALG

This ALG performs basic sanity checking on TCP packets. If it finds errors, it generates the following anomaly events and system log messages:

- TCP source or destination port zero
- TCP header length check failed
- TCP sequence number zero and no flags are set

- TCP sequence number zero and FIN/PSH/RST flags are set
- TCP FIN/RST or SYN(URG|FIN|RST) flags set

The TCP ALG performs the following steps:

1. When the router receives a SYN packet, the ALG creates TCP forward and reverse flows and groups them in a *conversation*. It tracks the TCP three-way handshake.
2. The SYN-defense mechanism tracks the TCP connection establishment state. It expects the TCP session to be established within a small time interval (currently 4 seconds). If the TCP three-way handshake is not established in that period, the session is terminated.
3. A keepalive mechanism detects TCP sessions with nonresponsive endpoints.
4. ICMP errors are allowed only if there is a flow that matches the selector information specified in the ICMP data.

Basic UDP ALG

This ALG performs basic sanity checking on UDP headers. If it finds errors. it generates the following anomaly events and system log messages:

- UDP source or destination port 0
- UDP header length check failed

The UDP ALG performs the following steps:

1. When it receives the first packet, the ALG creates bidirectional flows to accept forward and reverse UDP session traffic.
2. If the session is idle for more than the maximum allowed idle time (the default is 30 seconds), the flows are deleted.
3. ICMP errors are allowed only if there is a flow that matches the selector information specified in the ICMP data.

BOOTP

The Bootstrap Protocol client retrieves its networking information from a server across the network. It sends out a general broadcast message to request the information, which is returned by the Bootstrap Protocol server. For the protocol specification, see **ftp://ftp.isi.edu/in-notes/rfc951.txt**.

Stateful firewall support requires that you configure the BOOTP ALG on UDP server port 67 and client port 68. If the client sends a broadcast message, you should configure the broadcast address in the **from** statement of the service rule. NAT is not performed on the BOOTP traffic, even if the NAT rule matches the traffic. If the BOOTP relay feature is activated on the router, the remote BOOTP server is assumed to assign addresses for clients masked by NAT translation.

DCE RPC Services

DCE RPC services are mainly used by Microsoft applications. The ALG uses well-known TCP port 135 for port mapping services and uses the Universal Unique Identifier (UUID)

instead of the program number to identify protocols. The main application-based DCE RPC is the Microsoft Exchange Protocol.

Support for stateful firewall and NAT services requires that you configure the DCE RPC portmap ALG on TCP port 135. The DCE RPC ALG uses the TCP protocol with application-specific UUIDs.

ONC RPC Services

ONC RPC services function similarly to DCE RCP services. However, the ONC RPC ALG uses TCP/UDP port 111 for port mapping services and uses the program number to identify protocols rather than the UUID.

Support for stateful firewall and NAT services requires that you configure the ONC RPC portmap ALG on TCP port 111. The ONC RPC ALG uses the TCP protocol with application-specific program numbers.

FTP

FTP is the File Transfer Protocol, specified in RFC 959. In addition to the main control connection, data connections are also made for any data transfer between the client and the server, and the host, port, and direction are negotiated through the control channel.

For non-passive-mode FTP, the Junos stateful firewall service scans the client-to-server application data for the PORT command, which provides the IP address and port number to which the server connects. For passive-mode FTP, the Junos stateful firewall service scans the client-to-server application data for the PASV command and then scans the server-to-client responses for the 227 response, which contains the IP address and port number to which the client connects.

There is an additional complication: FTP represents these addresses and port numbers in ASCII. As a result, when addresses and ports are rewritten, the TCP sequence number might be changed, and thereafter the NAT service needs to maintain this delta in SEQ and ACK numbers by performing sequence NAT on all subsequent packets.

Support for stateful firewall and NAT services requires that you configure the FTP ALG on TCP port 21 to enable the FTP control protocol. The ALG performs the following tasks:

- Automatically allocates data ports and firewall permissions for dynamic data connection
- Creates flows for the dynamically negotiated data connection
- Monitors the control connection in both active and passive modes
- Rewrites the control packets with the appropriate NAT address and port information

ICMP

The Internet Control Message Protocol (ICMP) is defined in RFC 792. The Junos stateful firewall service allows ICMP messages to be filtered by specific type or specific type code value. ICMP error packets that lack a specifically configured type and code are matched against any existing flow in the opposite direction to check for the legitimacy of the error packet. ICMP error packets that pass the filter matching are subject to NAT translation.

The ICMP ALG always tracks ping traffic statefully using the ICMP sequence number. Each echo reply is forwarded only if there is an echo request with the corresponding sequence number. For any ping flow, only 20 echo requests can be forwarded without receiving an echo reply. When you configure dynamic NAT, the PING packet identifier is translated to allow additional hosts in the NAT pool to use the same identifier.

Support for stateful firewall and NAT services requires that you configure the ICMP ALG if the protocol is needed. You can configure the ICMP type and code for additional filtering.

NetShow

The Microsoft protocol ms-streaming is used by NetShow, the Microsoft media server. This protocol supports several transport protocols: TCP, UDP, and HTTP. The client starts a TCP connection on port 1755 and sends the PORT command to the server. The server then starts UDP on that port to the client. Support for stateful firewall and NAT services requires that you configure the NetShow ALG on UDP port 1755.

RPC and RPC Portmap Services

The Remote Procedure Call (RPC) ALG uses well-known ports TCP 111 and UDP 111 for port mapping, which dynamically assigns and opens ports for RPC services. The RPC Portmap ALG keeps track of port requests and dynamically opens the firewall for these requested ports. The RPC ALG can further restrict the RPC protocol by specifying allowed program numbers.

The ALG includes the RPC services listed in Table 1 on page 7:

Table 1: Supported RPC Services

Name	Description	Comments
rpc-mountd	Network File Server (NFS) mount daemon for details, see the UNIX man page for **rpc.mountd(8)**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050).
rpc-nfsprog	Used as part of NFS. For details, see RFC 1094. See also RFC1813 for NFS v3.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050).
rpc-nisplus	Network Information Service Plus (NIS+), designed to replace NIS; it is a default naming service for Sun Solaris and is not related to the old NIS. No protocol information is available.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050).
rpc-nlockmgr	Network lock manager.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-nlockmgr** service can be allowed or blocked based on RPC program 100021.
rpc-pcnfsd	Kernel statistics server. For details, see the UNIX man pages for **rstatd** and **rpc.rstatd**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-rstat** service can be allowed or blocked based on RPC program 150001.

Table 1: Supported RPC Services *(continued)*

Name	Description	Comments
rpc-rwall	Used to write a message to users; for details, see the UNIX man page for **rpc.rwalld**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-rwall** service can be allowed or blocked based on RPC program 150008.
rpc-ypbind	NIS binding process. For details, see the UNIX man page for **ypbind**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-ypbind** service can be allowed or blocked based on RPC program 100007.
rpc-yppasswd	NIS password server. For details, see the UNIX man page for **yppasswd**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-yppasswd** service can be allowed or blocked based on RPC program 100009.
rpc-ypserv	NIS server. For details, see the UNIX man page for **ypserv**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-ypserv** service can be allowed or blocked based on RPC program 100004.
rpc-ypupdated	Network updating tool.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-ypupdated** service can be allowed or blocked based on RPC program 100028.
rpc-ypxfrd	NIS map transfer server. For details, see the UNIX man page for **rpc.ypxfrd**.	The base support is RPC v2 and the port mapper service on port 111 (see RFC 1050). Once the RPC program table is built, **rpc-ypxfrd** service can be allowed or blocked based on RPC program 100069.

Support for stateful firewall and NAT services that use port mapping requires that you configure the RPC portmap ALG on TCP/UDP destination port 111 and the RPC ALG for both TCP and UDP. You can specify one or more **rpc-program-number** values to further restrict allowed RPC protocols.

RTSP

The Real-Time Streaming Protocol (RTSP) controls the delivery of data with real-time properties such as audio and video. The streams controlled by RTSP may use RTP, but it is not required. Media may be transmitted on the same RTSP control stream. This is an HTTP-like text-based protocol, but client and server maintain session information. A session is established using the SETUP message and terminated using the TEARDOWN message. The transport (the media protocol, address, and port numbers) is negotiated in the setup and the setup-response.

Support for stateful firewall and NAT services requires that you configure the RTSP ALG for TCP port 554.

The ALG monitors the control connection, opens flows dynamically for media (RTP/RTSP) streams, and performs NAT address and port rewrites.

SMB

Server message block (SMB) is a popular PC protocol that allows sharing of files, disks, directories, printers, and in some cases, COM ports across a network. SMB is a client/server, request-response-based protocol. Though there are some exceptions to this, most of the communication takes place using the request reply paradigm. Servers make file systems and resources available to clients on the network. Clients can send commands (**smbs**) to the server that allow them to access these shared resources. SMB can run over multiple protocols, including TCP/IP, NetBEUI, and IPX/SPX. In almost all cases, the NetBIOS interface is used. Microsoft is trying to rename SMB-based networking to Windows Networking and the protocol to CIFS. The SMB protocol is undocumented, although there is a public CIFS group. For more information, refer to the following link on CIFS: ftp://ftp.microsoft.com/developr/drg/CIFS/.

The SMB name service uses well-known UDP and TCP port 137, without requiring a special ALG. For NetBIOS data tunneled through UDP port 138 or TCP port 139, you must configure the NetBIOS ALG. Support for stateful firewall and NAT services requires that you configure the NetBIOS ALG on UDP port 138 and TCP port 139. For SMB name services, both TCP and UDP port 137 must be opened, without a special ALG.

SNMP

SNMP is a communication protocol for managing TCP/IP networks, including both individual network devices and aggregated devices. The protocol is defined by RFC 1157. SNMP runs on top of UDP.

The Junos stateful firewall service implements the SNMP ALG to inspect the SNMP type. SNMP does not enforce stateful flow. Each SNMP type needs to be specifically enabled. Full SNMP support of stateful firewall services requires that you configure the SNMP ALG on UDP port 161. This enables the SNMP ***get*** and **get-next** commands, as well as their response traffic in the reverse direction: UDP port 161 enables the SNMP **get-response** command. If SNMP traps are permitted, you can configure them on UDP port 162, enabling the SNMP **trap** command.

SQLNet

The SQLNet protocol is used by Oracle SQL servers to execute SQL commands from clients, including load balancing and application-specific services.

Support of stateful firewall and NAT services requires that you configure the SQLNet ALG for TCP port 1521.

The ALG monitors the control packets, opens flows dynamically for data traffic, and performs NAT address and port rewrites.

TFTP

The Trivial File Transfer Protocol (TFTP) is specified in RFC 1350. The initial TFTP requests are sent to UDP destination port 69. Additional flows can be created to **get** or **put** individual files. Support of stateful firewall and NAT services requires that you configure the TFTP ALG for UDP destination port 69.

Traceroute

Traceroute is a tool for displaying the route that packets take to a network host. It uses the IP TTL field to trigger ICMP time-exceeded messages from routers or gateways. It sends UDP datagrams to destination ports that are believed to be not in use; destination ports are numbered using the formula: + *n*hops – 1. The default base port is 33434. To support traceroute through the firewall, two types of traffic must be passed through:

1. UDP probe packets (UDP destination port > 33000, IP TTL < 30)
2. ICMP response packets (ICMP type time-exceeded)

When NAT is applied, the IP address and port within the ICMP error packet also need to be changed.

Support of stateful firewall and NAT services requires you to configure the Traceroute ALG for UDP destination port 33434 to 33450. In addition, you can configure the TTL threshold to prevent UDP flood attacks with large TTL values.

UNIX Remote-Shell Services

Three protocols form the basis for UNIX remote-shell services:

Exec—Remote command execution; enables a user on the client system to execute a command on the remote system. The first command from client (**rcmd**) to server (**rshd**) uses well-known TCP port 512. A second TCP connection can be opened at the request of **rcmd**. The client port number for the second connection is sent to the server as an ASCII string.

Login—Better known as **rlogin**; uses well-known TCP port 513. For details, see RFC 1282. No special firewall processing is required.

Shell—Remote command execution; enables a user on the client system to execute a command on the remote system. The first command from client (**rcmd**) to server (**rshd**) uses well-known TCP port 514. A second TCP connection can be opened at the request of **rcmd**. The client port number for the second connection is sent to the server as an ASCII string.

Support of stateful firewall services requires that you configure the Exec ALG on TCP port 512, the Login ALG on TCP port 513, and the Shell ALG on TCP port 514. NAT remote-shell services require that any dynamic source port assigned be within the port range 512 to 1023. If you configure a NAT pool, this port range is reserved exclusively for remote shell applications.

Verifying the Output of ALG Sessions

This section contains examples of successful output from ALG sessions and information on system log configuration. You can compare the results of your sessions to check whether the configurations are functioning correctly.

FTP Example

This example analyzes the output during an active FTP session. It consists of four different flows; two are control flows and two are data flows. The example consists of the following parts:

Sample Output

The following is a complete sample output from the **show services stateful-firewall conversations application-protocol ftp** operational mode command:

```
user@host>show services stateful-firewall conversations application-protocol ftp
Interface: ms-1/3/0, Service set: CLBJI1-AAF001
Conversation: ALG protocol: ftp
  Number of initiators: 2, Number of responders: 2
Flow                                                 State    Dir       Frm count
TCP           1.1.79.2:14083 ->        2.2.2.2:21    Watch    I              13
    NAT source        1.1.79.2:14083   ->   194.250.1.237:50118
TCP           1.1.79.2:14104 ->        2.2.2.2:20    Forward  I               3
    NAT source        1.1.79.2:14104   ->   194.250.1.237:50119
TCP            2.2.2.2:21    ->  194.250.1.237:50118 Watch    O              12
    NAT dest     194.250.1.237:50118   ->        1.1.79.2:14083
TCP            2.2.2.2:20    ->  194.250.1.237:50119 Forward  O               5
    NAT dest     194.250.1.237:50119   ->        1.1.79.2:14104
```

For each flow, the first line shows flow information, including protocol (TCP), source address, source port, destination address, destination port, flow state, direction, and frame count.

- The state of a flow can be **Watch**, **Forward**, or **Drop**:
  - A **Watch** flow state indicates that the control flow is monitored by the ALG for information in the payload. NAT processing is performed on the header and payload as needed.
  - A **Forward** flow forwards the packets without monitoring the payload. NAT is performed on the header as needed.
  - A **Drop** flow drops any packet that matches the 5 tuple.

- The frame count (**Frm count**) shows the number of packets that were processed on that flow.

The second line shows the NAT information.

- **source** indicates source NAT.
- **dest** indicates destination NAT.
- The first address and port in the NAT line are the original address and port being translated for that flow.
- The second address and port in the NAT line are the translated address and port for that flow.

FTP System Log Messages

System log messages are generated during an FTP session. For more information about system logs, see “System Log Messages” on page 15.

The following system log messages are generated during creation of the FTP control flow:

- Rule Accept system log:

  Oct 27 11:42:54 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_RULE_ACCEPT: proto 6 (TCP) application: ftp, fe-3/3/3.0:1.1.1.2:4450 -> 2.2.2.2:21, Match SFW accept rule-set:, rule: ftp, term: 1

- Create Accept Flow system log:

  Oct 27 11:42:54 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_CREATE_ACCEPT_FLOW: proto 6 (TCP) application: ftp, fe-3/3/3.0:1.1.1.2:4450 -> 2.2.2.2:21, creating forward or watch flow

- System log for data flow creation:

  Oct 27 11:43:30 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_FTP_ACTIVE_ACCEPT: proto 6 (TCP) application: ftp, so-2/1/2.0:2.2.2.2:20 -> 1.1.1.2:50726, Creating FTP active mode forward flow

Analysis

Control Flows

The control flows are established after the three-way handshake is complete.

- Control flow from FTP client to FTP server. TCP destination port is 21.

```
TCP           1.1.79.2:14083 ->        2.2.2.2:21    Watch    I
     13
    NAT source        1.1.79.2:14083   ->   194.250.1.237:50118
```

- Control flow from FTP server to FTP client. TCP source port is 21.

```
TCP            2.2.2.2:21    ->  194.250.1.237:50118 Watch    O
     12
    NAT dest     194.250.1.237:50118   ->        1.1.79.2:14083
```


Data Flows

A data port of 20 is negotiated for data transfer during the course of the FTP control protocol. These two flows are data flows between the FTP client and the FTP server:

```
TCP           1.1.79.2:14104 ->        2.2.2.2:20    Forward  I              3
    NAT source        1.1.79.2:14104   ->   194.250.1.237:50119
TCP            2.2.2.2:20    ->  194.250.1.237:50119 Forward  O              5
    NAT dest     194.250.1.237:50119   ->        1.1.79.2:14104
```

Troubleshooting Questions

1. How do I know if the FTP ALG is active?
   - The ALG protocol field in the conversation should display **ftp**.
   - There should be a valid frame count (**Frm count**) in the control flows.
   - A valid frame count in the data flows indicates that data transfer has taken place.
2. What do I need to check if the FTP connection is established but data transfer does not take place?
   - Most probably, the control connection is up, but the data connection is down.
   - Check the conversations output to determine whether both the control and data flows are present.
3. How do I interpret each flow? What does each flow mean?
   - FTP control flow initiator flow—Flow with destination port 21
   - FTP control flow responder flow—Flow with source port ;21
   - FTP data flow initiator flow—Flow with destination port 20
   - FTP data flow responder flow—Flow with source port 20

RTSP ALG Example

The following is an example of an RTSP conversation. The application uses the RTSP protocol for control connection. Once the connection is set up, the media is sent using UDP protocol (RTP).

This example consists of the following:

- Sample Output on page 13
- Analysis on page 14
- Troubleshooting Questions on page 14

Sample Output

Here is the output from the **show services stateful-firewall conversations** operational mode command:

```
user@host# show services stateful-firewall conversations
Interface: ms-3/2/0, Service set: svc_set
Conversation: ALG protocol: rtsp
  Number of initiators: 5, Number of responders: 5
```

```
Flow                                                  State    Dir       Frm count
TCP           1.1.1.3:58795 ->        2.2.2.2:554   Watch    I             7
UDP           1.1.1.3:1028  ->        2.2.2.2:1028  Forward  I             0
UDP           1.1.1.3:1029  ->        2.2.2.2:1029  Forward  I             0
UDP           1.1.1.3:1030  ->        2.2.2.2:1030  Forward  I             0
UDP           1.1.1.3:1031  ->        2.2.2.2:1031  Forward  I             0
TCP           2.2.2.2:554   ->        1.1.1.3:58795 Watch    O             5
UDP           2.2.2.2:1028  ->        1.1.1.3:1028  Forward  O             6
UDP           2.2.2.2:1029  ->        1.1.1.3:1029  Forward  O             0
UDP           2.2.2.2:1030  ->        1.1.1.3:1030  Forward  O             3
UDP           2.2.2.2:1031  ->        1.1.1.3:1031  Forward  O             0
```

Analysis

An RTSP conversation should consist of TCP flows corresponding to the RTSP control connection. There should be two flows, one in each direction, from client to server and from server to client:

```
TCP           1.1.1.3:58795 ->        2.2.2.2:554   Watch    I             7
TCP           2.2.2.2:554   ->        1.1.1.3:58795 Watch    O             5
```

- The RTSP control connection for the initiator flow is sent from destination port 554.
- The RTSP control connection for the responder flow is sent from source port 554.

The UDP flows correspond to RTP media sent over the RTSP connection.

Troubleshooting Questions

1. Media does not work when the RTSP ALG is configured. What do I do?
   - Check RTSP conversations to see whether both TCP and UDP flows exist.
   - The ALG protocol should be displayed as **rtsp**.

   NOTE: The state of the flow is displayed as Watch, because the ALG processing is taking place and the client is essentially "watching" or processing payload corresponding to the application. For FTP and RTSP ALG flows, the control connections are always Watch flows.

2. How do I check for ALG errors?
   - You can check for errors by issuing the following command. Each ALG has a separate field for ALG packet errors.

     ```
     user@host# show services stateful-firewall statistics extensive

     Interface: ms-3/2/0
       Service set: svc_set
         New flows:
           Accepts: 1347, Discards: 0, Rejects: 0
         Existing flows:
           Accepts: 144187, Discards: 0, Rejects: 0
         Drops:
           IP option: 0, TCP SYN defense: 0
           NAT ports exhausted: 0
         Errors:
           IP: 0, TCP: 276
     ```

```
  UDP: 0, ICMP: 0
  Non-IP packets: 0, ALG: 0
IP errors:
  IP packet length inconsistencies: 0
  Minimum IP header length check failures: 0
  Reassembled packet exceeds maximum IP length: 0
  Illegal source address: 0
  Illegal destination address: 0
  TTL zero errors: 0, Illegal IP protocol number (0 or 255): 0
  Land attack: 0
  Non-IPv4 packets: 0, Bad checksum: 0
  Illegal IP fragment length: 0
  IP fragment overlap: 0
  IP fragment reassembly timeout: 0
  Unknown: 0
TCP errors:
  TCP header length inconsistencies: 0
  Source or destination port number is zero: 0
  Illegal sequence number and flags combinations: 0
  SYN attack (multiple SYN messages seen for the same flow): 276
  First packet not a SYN message: 0
  TCP port scan (TCP handshake, RST seen from server for SYN): 0
  Bad SYN cookie response: 0
UDP errors:
  IP data length less than minimum UDP header length (8 bytes): 0
  Source or destination port number is zero: 0
  UDP port scan (ICMP error seen for UDP flow): 0
ICMP errors:
  IP data length less than minimum ICMP header length (8 bytes): 0
  ICMP error length inconsistencies: 0
  Duplicate ping sequence number: 0
  Mismatched ping sequence number: 0
ALG errors:
  BOOTP: 0, DCE-RPC: 0, DCE-RPC portmap: 0
  DNS: 0, Exec: 0, FTP: 0
  ICMP: 0
  Login: 0, NetBIOS: 0, NetShow: 0
  RPC: 0, RPC portmap: 0
  RTSP: 0, Shell: 0
  SNMP: 0, SQLNet: 0, TFTP: 0
  Traceroute: 0
```

System Log Messages

Enabling system log generation and checking the system log are also helpful for ALG flow analysis. This section contains the following:

- System Log Configuration on page 15
- System Log Output on page 16

System Log Configuration

You can configure the enabling of system log messages at a number of different levels in the Junos OS CLI. As shown in the following sample configurations, the choice of level depends on how specific you want the event logging to be and what options you want to include. For details on the configuration options, see the *Junos OS System Basics Configuration Guide* (system level) or the *Junos OS Services Interfaces Configuration Guide* (all other levels).

1. At the topmost global level:

```
user@host# show system syslog
file messages {
    any any;
}
```

2. At the service set level:

```
user@host# show services service-set svc_set
syslog {
    host local {
        services any;
    }
}
stateful-firewall-rules allow_rtsp;
interface-service {
    service-interface ms-3/2/0;
}
```

3. At the service rule level:

```
user@host# show services stateful-firewall rule allow_rtsp
match-direction input-output;
term 0 {
    from {
        applications junos-rtsp;
    }
    then {
        accept;
        syslog;
    }
}
```

System Log Output

System log messages are generated during flow creation, as shown in the following examples:

The following system log message indicates that the ASP matched an accept rule:

```
Oct 25 16:11:37 (FPC Slot 3, PIC Slot 2) {svc_set}[FWNAT]: ASP_SFW_RULE_ACCEPT:
  proto 6 (TCP) application: rtsp, ge-2/0/1.0:1.1.1.2:35595 -> 2.2.2.2:554, Match SFW accept
  rule-set: , rule: allow_rtsp, term: 0
```

For a complete listing of system log messages, see the *Junos OS System Log Messages Reference*.

Junos Default Groups

The Junos OS provides a default, hidden configuration group called **junos-defaults** that is automatically applied to the configuration of your router. The **junos-defaults** group contains preconfigured statements that contain predefined values for common applications. Some of the statements must be referenced to take effect, such as applications like FTP or Telnet. Other statements are applied automatically, such as terminal settings. All of the preconfigured statements begin with the reserved name **junos-**.

NOTE: You can override the Junos default configuration values, but you cannot delete or edit them. If you delete a configuration, the defaults return when a new configuration is added.

You cannot use the **apply-groups** statement with the Junos defaults group.

To view the full set of available preset statements from the Junos default group, issue the **show groups junos-defaults** configuration mode command. The following example displays a partial list of Junos default groups that use application protocols (ALGs).

```
user@host# show groups junos-defaults
... output for other groups defined at the  [edit groups junos-defaults] hierarchy level ...
applications {
    # File Transfer Protocol
    application junos-ftp {
        application-protocol ftp;
        protocol tcp;
        destination-port 21;
    }
    # Trivial File Transfer Protocol
    application junos-tftp {
        application-protocol tftp;
        protocol udp;
        destination-port 69;
    }
    # RPC port mapper on TCP
    application junos-rpc-portmap-tcp {
        application-protocol rpc-portmap;
        protocol tcp;
        destination-port 111;
    }
    # RPC port mapper on UDP
    application junos-rpc-portmap-udp {
        application-protocol rpc-portmap;
        protocol udp;
        destination-port 111;
    }
    # IP Protocol
    application junos-ip {
        application-protocol ip;
    }
    # remote exec
    application junos-rexec {
        application-protocol exec;
        protocol tcp;
        destination-port 512;
    }
    # remote login
    application junos-rlogin {
        application-protocol login;
        protocol tcp;
        destination-port 513;
    }
```

```
# remote shell
application junos-rsh {
   application-protocol shell;
   protocol tcp;
   destination-port 514;
}
# Real-Time Streaming Protocol
application junos-rtsp {
   application-protocol rtsp;
   protocol tcp;
   destination-port 554;
}
# Oracle SQL servers use this protocol to execute SQL commands
# from clients, load balance, use application-specific servers, and so on.
application junos-sqlnet {
   application-protocol sqlnet;
   protocol tcp;
   destination-port 1521;
}
# H.323 Protocol for audio/video conferencing
protocol tcp;
   destination-port 1720;
}
# Internet Inter-ORB Protocol is used for CORBA applications.
# The ORB protocol in Java virtual machine uses port 1975 as a default.
protocol tcp;
   destination-port 1975;
}
# Internet Inter-ORB Protocol is used for CORBA applications.
# ORBIX is a CORBA framework from Iona Technologies that uses
# port 3075 as a default.
protocol tcp;
   destination-port 3075;
}
# This was the original RealPlayer protocol.
# RTSP is more widely used by RealPlayer,
protocol tcp;
   destination-port 7070;
}
# Traceroute application
application junos-traceroute {
   application-protocol traceroute;
   protocol udp;
   destination-port 33435-33450;
   ttl-threshold 30;
}
# Traceroute application that stops at device supporting firewall
# (packets with ttl > 1 will be discarded).
application junos-traceroute-ttl-1 {
   application-protocol traceroute;
   protocol udp;
   destination-port 33435-33450;
   ttl-threshold 1;
}
# The full range of known RPC programs using UDP.
# Specific program numbers are assigned to certain applications.
```

```
application junos-rpc-services-udp {
   application-protocol rpc;
   protocol udp;
   rpc-program-number 100001-400000;
}
# The full range of known RPC programs using TCP.
# Specific program numbers are assigned to certain applications.
application junos-rpc-services-tcp {
   application-protocol rpc;
   protocol tcp;
   rpc-program-number 100001-400000;
}
# All ICMP traffic
# This can be made more restrictive by specifying ICMP type and code.
application junos-icmp-all {
   application-protocol icmp;
}
# ICMP ping; the echo reply is allowed upon return.
application junos-icmp-ping {
   application-protocol icmp;
   icmp-type echo-request;
}
# Protocol used by Windows Media Server and Windows Media Player
application junos-netshow {
   application-protocol netshow;
   protocol tcp;
   destination-port 1755;
}
# NetBIOS, the networking protocol used on Windows networks;
# includes name service port, both UDP and TCP.
application junos-netbios-name-udp {
   application-protocol netbios;
   protocol udp;
   destination-port 137;
}
application junos-netbios-name-tcp {
   protocol tcp;
   destination-port 137;
}
# NetBIOS, the networking protocol used on Windows networks;
# includes datagram service port.
application junos-netbios-datagram {
   application-protocol netbios;
   protocol udp;
   destination-port 138;
}
# NetBIOS, the networking protocol used on Windows networks;
# includes session service port.
application junos-netbios-session {
   protocol tcp;
   destination-port 139;
}
# DCE-RPC port mapper on TCP
application junos-dce-rpc-portmap {
   application-protocol dce-rpc-portmap;
   protocol tcp;
```

```
    destination-port 135;
}
# MS Exchange requires these three UUID values.
application junos-dcerpc-endpoint-mapper-service {
    application-protocol dce-rpc;
    protocol tcp;
    uuid e1af8308-5d1f-11c9-91a4-08002b14a0fa;
}
application junos-ssh {
    protocol tcp;
    destination-port 22;
}
application junos-telnet {
    protocol tcp;
    destination-port 23;
}
application junos-smtp {
    protocol tcp;
    destination-port 25;
}
application junos-dns-udp {
    protocol udp;
    destination-port 53;
}
application junos-dns-tcp {
    protocol tcp;
    destination-port 53;
}
application junos-tacacs {
    protocol tcp;
    destination-port 49;
}
# TACACS Database Service
application junos-tacacs-ds {
    protocol tcp;
    destination-port 65;
}
application junos-dhcp-client {
    protocol udp;
    destination-port 68;
}
application junos-dhcp-server {
    protocol udp;
    destination-port 67;
}
application junos-bootpc {
    protocol udp;
    destination-port 68;
}
application junos-bootps {
    protocol udp;
    destination-port 67;
}
application junos-http {
    protocol tcp;
    destination-port 80;
```

```
}
application junos-https {
   protocol tcp;
   destination-port 443;
}
# " junos-algs-outbound" defines a set of all applications
# requiring an ALG. Useful for defining a rule for an untrusted
# network to allow trusted network users to use all the
# Junos-supported ALGs initiated from the trusted network.
application-set junos-algs-outbound {
   application junos-ftp;
   application junos-tftp;
   application junos-rpc-portmap-tcp;
   application junos-rpc-portmap-udp;
   application junos-snmp-get;
   application junos-snmp-get-next;
   application junos-snmp-response;
   application junos-snmp-trap;
   application junos-rexec;
   application junos-rlogin;
   application junos-rsh;
   application junos-rtsp;
   application junos-sqlnet;
   application junos-traceroute;
   application junos-rpc-services-udp;
   application junos-rpc-services-tcp;
   application junos-icmp-all;
   application junos-netshow;
   application junos-netbios-name-udp;
   application junos-netbios-datagram;
   application junos-dce-rpc-portmap;
   application junos-dcerpc-msexchange-directory-rfr;
   application junos-dcerpc-msexchange-information-store;
   application junos-dcerpc-msexchange-directory-nsp;
}
# " junos-management-inbound" represents the group of applications
# that might need access to the trusted network from the untrusted
# network for management purposes.
# The set is intended for a UI to display management choices.
# NOTE: It is not recommended that you use the entire set directly in
# a firewall rule and open up firewall to all of these
# applications. Also, you should always specify the source
# and destination prefixes when using each application.
application-set junos-management-inbound {
   application junos-snmp-get;
   application junos-snmp-get-next;
   application junos-snmp-response;
   application junos-snmp-trap;
   application junos-ssh;
   application junos-telnet;
   application junos-http;
   application junos-https;
   application junos-xnm-ssl;
   application junos-xnm-clear-text;
   application junos-icmp-ping;
   application junos-traceroute-ttl-1;
```

```
    }
  }
  }
  }
```

To reference statements available from the **junos-defaults** group, include the selected **junos-*default-name*** statement at the applicable hierarchy level. To configure application protocols, see “Configuring Application Protocol Properties” on page 27; for details about a specific protocol, see “ALG Descriptions” on page 4.

Examples: Referencing the Preset Statement from the Junos Default Group

The following example is a preset statement from the Junos default groups that is available for FTP in a stateful firewall:

```
[edit]
groups {
  junos-defaults {
    applications {
      application junos-ftp { # Use FTP default configuration
      application-protocol ftp;
      protocol tcp;
      destination-port 21;
    }
  }
}
```

To reference a preset Junos default statement from the Junos default groups, include the **junos-*default-name*** statement at the applicable hierarchy level. For example, to reference the Junos default statement for FTP in a stateful firewall, include the **junos-ftp** statement at the **[edit services stateful-firewall rule *rule-name* term *term-name* from applications]** hierarchy level.

```
[edit]
services {
  stateful-firewall {
    rule my-rule {
      term my-term {
        from {
          applications junos-ftp; #Reference predefined statement, junos-ftp,
        }
      }
    }
  }
}
```

The following example shows configuration of the default Junos IP ALG:

```
[edit]
services {
  stateful-firewall {
    rule r1 {
      match-direction input;
      term t1 {
        from {
          applications junos-ip;
        }
```



```
            then {
              accept;
              syslog;
            }
          }
        }
      }
    }
```

If you configure the IP ALG in the stateful firewall rule, it is matched by any IP traffic, but if there is any other more specific application that matches the same traffic, the IP ALG will not be matched. For example, in the following configuration, both the ICMP ALG and the IP ALG are configured, but traffic is matched for ICMP packets, because it is the more specific match.

```
[edit]
services {
  stateful-firewall {
    rule r1 {
      match-direction input;
      term t1 {
        from {
          applications [ junos-ip junos-icmp-all ];
        }
        then {
          accept;
          syslog;
        }
      }
    }
  }
}
```

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Configuring Application Protocol Properties

To configure application properties, include the **application** statement at the **[edit applications]** hierarchy level:

```
[edit applications]
application application-name {
    application-protocol protocol-name;
    destination-port port-number;
    icmp-code value;
    icmp-type value;
    inactivity-timeout value;
    protocol type;
    rpc-program-number number;
    snmp-command command;
    source-port port-number;
    ttl-threshold value;
    uuid hex-value;
}
```

You can group application objects by configuring the **application-set** statement; for more information, see “Configuring Application Sets” on page 36.

This section includes the following tasks for configuring applications:

- Configuring the TTL Threshold on page 36
- Configuring a Universal Unique Identifier on page 36

Configuring an Application Protocol

The **application-protocol** statement allows you to specify which of the supported application protocols (ALGs) to configure and include in an application set for service processing. To configure application protocols, include the **application-protocol** statement at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
application-protocol protocol-name;
```

Table 2 on page 28 shows the list of supported protocols. For more information about specific protocols, see "ALG Descriptions" on page 4.

Table 2: Application Protocols Supported by Services Interfaces

Protocol Name	CLI Value	Comments
Bootstrap protocol (BOOTP)	**bootp**	Supports BOOTP and dynamic host configuration protocol (DHCP).
Distributed Computing Environment (DCE) remote procedure call (RPC)	**dce-rpc**	Requires the **protocol** statement to have the value **udp** or **tcp**. Requires a **uuid** value. You cannot specify **destination-port** or **source-port** values.
DCE RPC portmap	**dce-rpc-portmap**	Requires the **protocol** statement to have the value **udp** or **tcp**. Requires a **destination-port** value.
Domain Name System (DNS)	**dns**	Requires the **protocol** statement to have the value **udp**. This application protocol closes the DNS flow as soon as the DNS response is received.
Exec	**exec**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** value.
FTP	**ftp**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** value.
Internet Control Message Protocol (ICMP)	**icmp**	Requires the **protocol** statement to have the value **icmp** or to be unspecified.
IP	**ip**	–
Login	**login**	–
NetBIOS	**netbios**	Requires the **protocol** statement to have the value **udp** or to be unspecified. Requires a **destination-port** value.
NetShow	**netshow**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** value.
Real-Time Streaming Protocol (RTSP)	**rtsp**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** value.

Table 2: Application Protocols Supported by Services Interfaces *(continued)*

Protocol Name	CLI Value	Comments
RPC User Datagram Protocol (UDP) or TCP	**rpc**	Requires the **protocol** statement to have the value **udp** or **tcp**. Requires a **rpc-program-number** value. You cannot specify **destination-port** or **source-port** values.
RPC port mapping	**rpc-portmap**	Requires the **protocol** statement to have the value **udp** or **tcp**. Requires a **destination-port** value.
Shell	**shell**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** value.
SNMP	**snmp**	Requires the **protocol** statement to have the value **udp** or to be unspecified. Requires a **destination-port** value.
SQLNet	**sqlnet**	Requires the **protocol** statement to have the value **tcp** or to be unspecified. Requires a **destination-port** or **source-port** value.
Trace route	**traceroute**	Requires the **protocol** statement to have the value **udp** or to be unspecified. Requires a **destination-port** value.
Trivial FTP (TFTP)	**tftp**	Requires the **protocol** statement to have the value **udp** or to be unspecified. Requires a **destination-port** value.

NOTE: You can configure application-level gateways (ALGs) for ICMP and trace route under stateful firewall, NAT, or CoS rules when twice NAT is configured in the same service set. These ALGs cannot be applied to flows created by the Packet Gateway Controller Protocol (PGCP). Twice NAT does not support any other ALGs. NAT applies only the IP address and TCP or UDP headers, but not the payload.

For more information about configuring twice NAT, see Network Address Translation.

Configuring the Network Protocol

The **protocol** statement allows you to specify which of the supported network protocols to match in an application definition. To configure network protocols, include the **protocol** statement at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
protocol type;
```

You specify the protocol type as a numeric value; for the more commonly used protocols, text names are also supported in the command-line interface (CLI). Table 3 on page 30 shows the list of the supported protocols.

Table 3: Network Protocols Supported by Services Interfaces

Network Protocol Type	CLI Value	Comments
IP Security (IPsec) authentication header (AH)	**ah**	–
External Gateway Protocol (EGP)	**egp**	–
IPsec Encapsulating Security Payload (ESP)	**esp**	–
Generic routing encapsulation (GR)	**gre**	–
ICMP	**icmp**	Requires an **application-protocol** value of **icmp**.
Internet Group Management Protocol (IGMP)	**igmp**	–
IP in IP	**ipip**	–
OSPF	**ospf**	–
Protocol Independent Multicast (PIM)	**pim**	–
Resource Reservation Protocol (RSVP)	**rsvp**	–
TCP	**tcp**	Requires a **destination-port** or **source-port** value unless you specify **application-protocol rcp** or **dce-rcp**.
UDP	**udp**	Requires a **destination-port** or **source-port** value unless you specify **application-protocol rcp** or **dce-rcp**.
Virtual Router Redundancy Protocol (VRRP)	**vrrp**	–

For a complete list of possible numeric values, see RFC 1700, *Assigned Numbers (for the Internet Protocol Suite)*.

NOTE: IP version 6 (IPv6) is not supported as a network protocol in application definitions.

By default, the twice NAT feature can affect IP, TCP, and UDP headers embedded in the payload of ICMP error messages. You can include the **protocol tcp** and **protocol udp** statements with the application statement for twice NAT configurations. For more information about configuring twice NAT, see Network Address Translation.

Configuring the ICMP Code and Type

The ICMP code and type provide additional specification, in conjunction with the network protocol, for packet matching in an application definition. To configure ICMP settings, include the **icmp-code** and **icmp-type** statements at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
icmp-code value;
icmp-type value;
```

You can include only one ICMP code and type value. The **application-protocol** statement must have the value **icmp**. Table 4 on page 31 shows the list of supported ICMP values.

Table 4: ICMP Codes and Types Supported by Services Interfaces

CLI Statement	Description
icmp-code	This value or keyword provides more specific information than **icmp-type**. Because the value's meaning depends upon the associated **icmp-type** value, you must specify **icmp-type** along with **icmp-code**. For more information, see the *Junos OS Policy Framework Configuration Guide*. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed). The keywords are grouped by the ICMP type with which they are associated: parameter-problem: **ip-header-bad** (0), **required-option-missing** (1) redirect: **redirect-for-host** (1), **redirect-for-network** (0), **redirect-for-tos-and-host** (3), **redirect-for-tos-and-net** (2) time-exceeded: **ttl-eq-zero-during-reassembly** (1), **ttl-eq-zero-during-transit** (0) unreachable: **communication-prohibited-by-filtering** (13), **destination-host-prohibited** (10), **destination-host-unknown** (7), **destination-network-prohibited** (9), **destination-network-unknown** (6), **fragmentation-needed** (4), **host-precedence-violation** (14), **host-unreachable** (1), **host-unreachable-for-TOS** (12), **network-unreachable** (0), **network-unreachable-for-TOS** (11), **port-unreachable** (3), **precedence-cutoff-in-effect** (15), **protocol-unreachable** (2), **source-host-isolated** (8), **source-route-failed** (5)
icmp-type	Normally, you specify this match in conjunction with the **protocol** match statement to determine which protocol is being used on the port. For more information, see the *Junos OS Policy Framework Configuration Guide*. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): **echo-reply** (0), **echo-request** (8), **info-reply** (16), **info-request** (15), **mask-request** (17), **mask-reply** (18), **parameter-problem** (12), **redirect** (5), **router-advertisement** (9), **router-solicit** (10), **source-quench** (4), **time-exceeded** (11), **timestamp** (13), **timestamp-reply** (14), or **unreachable** (3).

NOTE: If you configure an interface with an input firewall filter that includes a reject action and with a service set that includes stateful firewall rules, the router executes the input firewall filter before the stateful firewall rules are run on the packet. As a result, when the Packet Forwarding Engine sends an ICMP error message out through the interface, the stateful firewall rules might drop the packet because it was not seen in the input direction.

Possible workarounds are to include a forwarding-table filter to perform the reject action, because this type of filter is executed after the stateful firewall in the input direction, or to include an output service filter to prevent the locally generated ICMP packets from going to the stateful firewall service.

Configuring Source and Destination Ports

The TCP or UDP source and destination port provide additional specification, in conjunction with the network protocol, for packet matching in an application definition. To configure ports, include the **destination-port** and **source-port** statements at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
destination-port value;
source-port value;
```

You must define one source or destination port. Normally, you specify this match in conjunction with the **protocol** match statement to determine which protocol is being used on the port; for constraints, see Table 2 on page 28.

You can specify either a numeric value or one of the text synonyms listed in Table 5 on page 32.

Table 5: Port Names Supported by Services Interfaces

Port Name	Corresponding Port Number
afs	1483
bgp	179
biff	512
bootpc	68
bootps	67
cmd	514
cvspserver	2401
dhcp	67
domain	53

Table 5: Port Names Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
eklogin	2105
ekshell	2106
exec	512
finger	79
ftp	21
ftp-data	20
http	80
https	443
ident	113
imap	143
kerberos-sec	88
klogin	543
kpasswd	761
krb-prop	754
krbupdate	760
kshell	544
ldap	389
login	513
mobileip-agent	434
mobilip-mn	435
msdp	639
netbios-dgm	138
netbios-ns	137
netbios-ssn	139

Table 5: Port Names Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
nfsd	2049
nntp	119
ntalk	518
ntp	123
pop3	110
pptp	1723
printer	515
radacct	1813
radius	1812
rip	520
rkinit	2108
smtp	25
snmp	161
snmptrap	162
snpp	444
socks	1080
ssh	22
sunrpc	111
syslog	514
tacacs-ds	65
talk	517
telnet	23
tftp	69
timed	525

Table 5: Port Names Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
who	513
xdmcp	177
zephyr-clt	2103
zephyr-hm	2104

For more information about matching criteria, see the *Junos OS Policy Framework Configuration Guide*.

Configuring the Inactivity Timeout Period

You can specify a timeout period for application inactivity. If the software has not detected any activity during the duration, the flow becomes invalid when the timer expires. To configure a timeout period, include the **inactivity-timeout** statement at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
inactivity-timeout seconds;
```

The default value is 30 seconds. The value you configure for an application overrides any global value configured at the **[edit interfaces *interface-name* service-options]** hierarchy level; for more information, see Configuring Default Timeout Settings for Services Interfaces.

Configuring an SNMP Command for Packet Matching

You can specify an SNMP command setting for packet matching. To configure SNMP, include the **snmp-command** statement at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
snmp-command value;
```

The supported values are **get**, **get-next**, **set**, and **trap**. You can configure only one value for matching. The **application-protocol** statement at the **[edit applications application *application-name*]** hierarchy level must have the value **snmp**. For information about specifying the application protocol, see “Configuring an Application Protocol” on page 28.

Configuring an RPC Program Number

You can specify an RPC program number for packet matching. To configure an RPC program number, include the **rpc-program-number** statement at the **[edit applications application *application-name*]** hierarchy level:

```
[edit applications application application-name]
rpc-program-number number;
```

The range of values used for DCE or RPC is from 100,000 through 400,000. The **application-protocol** statement at the **[edit applications application *application-name*]**

hierarchy level must have the value **rpc**. For information about specifying the application protocol, see "Configuring an Application Protocol" on page 28.

Configuring the TTL Threshold

You can specify a trace route time-to-live (TTL) threshold value, which controls the acceptable level of network penetration for trace routing. To configure a TTL value, include the **ttl-threshold** statement at the **[edit applications application** *application-name***]** hierarchy level:

```
[edit applications application application-name]
ttl-threshold value;
```

The **application-protocol** statement at the **[edit applications application** *application-name***]** hierarchy level must have the value **traceroute**. For information about specifying the application protocol, see "Configuring an Application Protocol" on page 28.

Configuring a Universal Unique Identifier

You can specify a Universal Unique Identifier (UUID) for DCE RPC objects. To configure a UUID value, include the **uuid** statement at the **[edit applications application** *application-name***]** hierarchy level:

```
[edit applications application application-name]
uuid hex-value;
```

The **uuid** value is in hexadecimal notation. The **application-protocol** statement at the **[edit applications application** *application-name* hierarchy level must have the value **dce-rpc**. For information about specifying the application protocol, see "Configuring an Application Protocol" on page 28. For more information on UUID numbers, see **http://www.opengroup.org/onlinepubs/9629399/apdxa.htm**.

Configuring Application Sets

You can group the applications you have defined into a named object by including the **application-set** statement at the **[edit applications]** hierarchy level with an **application** statement for each application:

```
[edit applications]
  application-set application-set-name {
  application application;
}
```

For an example of a typical application set, see "Examples: Configuring Application Protocols" on page 45.

Configuring Juniper Service Framework – Application-Level Gateways, Rules, and Services Set

ALGs intercept and analyze specified traffic, allocate resources, and define dynamic policies to permit traffic to pass securely through a device. You may use JSF ALGs with the SFW and NAT.

To use JSF to run ALGs, you must configure the jservices-nat, jservices-alg, and jservices-sfw package at the hierarchy level. In addition, you must configure SFW rules and a services set with a Multiservice interface. This section includes the following tasks:

1. Configuring the JSF Application-Level Gateways Package on page 37
2. Configuring the Stateful Firewall with Application-Level Gateways on page 39
3. Configuring the NAT with ALG on page 40

Configuring the JSF Application-Level Gateways Package

To configure the JSF services:

1. In configuration mode, go to the following hierarchy level:

   user@host# edit chassis

2. In the hierarchy level, configure the FPC and PIC.

   [edit chassis]
   user@host# edit fpc *slot* pic *slot*

   In this example, the FPC is in slot 1 and the PIC is in slot 0:

   [edit chassis]
   user@host# edit fpc 1 pic 0

3. Configure the number of cores dedicated to run control functionality.

   [edit chassis fpc *slot* pic *slot*]
   user@host# set adaptive-services service-package extension-provider control-cores *control-cores*

   In this example, the number of control cores is 1.

   [edit chassis fpc 1 pic 0]
   user@host# set adaptive-services service-package extension-provider control-cores 1

4. Configure the number of processing cores dedicated to data.

   [edit chassis fpc *slot* pic *slot*]
   user@host# set adaptive-services service-package extension-provider data-cores *data-cores*

   In this example, the number of data cores is 7.

   [edit chassis fpc 1 pic 0]
   user@host# set adaptive-services service-package extension-provider data-cores 7

5. Configure the size of the object cache in MB. Only values in increments of 128 MB are allowed and the maximum value of object cache can be 1280 MB. On MS-100 the value is 512 MB. To configure the size of the cache:

   [edit chassis fpc *slot* pic *slot*]
   user@host# set adaptive-services service-package extension-provider object-cache-size *object-cache-size*

In this example, the size of the object cache is 1280 MB.

```
[edit chassis fpc 1 pic 0]
user@host# set adaptive-services service-package extension-provider
  object-cache-size 1280
```

6. Configure the size of the policy database in MB.

```
[edit chassis fpc slot pic slot]
user@host# set adaptive-services service-package extension-provider policy-db-size
  policy-db-size
```

In this example, the size of the policy database is 64 MB.

```
[edit chassis fpc 1 pic 0]
user@host# set adaptive-services service-package extension-provider policy-db-size
  64
```

7. Configure the package.

```
[edit chassis fpc slot pic slot]
user@host# set adaptive-services service-package extension-provider package
  package
```

In this example, the first package is **jservices-nat**, the second package is **jservices-alg**, and the third package is **jservices-sfw**.

```
[edit chassis fpc 1 pic 0]
user@host# set adaptive-services service-package extension-provider package
  jservices-nat
user@host# set adaptive-services service-package extension-provider package
  jservices-alg
user@host# set adaptive-services service-package extension-provider package
  jservices-sfw
```

8. Configure the extension provider system log, to enable PIC system logging to record or view system log messages:

```
[edit chassis fpc slot pic slot]
user@host# set adaptive-services service-package extension-provider syslog syslog
```

In this example **syslog** is set to **daemon any** and **external any**:

```
[edit chassis fpc 1 pic 0]
user@host# set adaptive-services service-package extension-provider syslog daemon
  any
[edit chassis fpc 1 pic 0]
user@host# set adaptive-services service-package extension-provider syslog external
  any
```

9. Verify the configuration.

```
[edit chassis]
user@host# show chassis
fpc 1 {
    pic 0 {
        adaptive-services {
            service-package {
                extension-provider {
                    control-cores 1;
                    data-cores 7;
                    object-cache-size 1280;
```



```
                        policy-db-size 64;
                        package jservices-nat;
                        package jservices-alg;
                        package jservices-sfw;
                        syslog {
                            daemon any;
                            external any;
                        }
                    }
                }
            }
        }
    }
```

10. Verify for ALG errors in the configuration.

```
host@user# run show services alg statistics

Interface name: ms-0/0/0

  RSH ALG statistics:

    Invalid packets recevied  : 0

    Packets dropped by ALG    : 0

    ALG parser errors         : 0

    Packets freed by ALG      : 0

  DNS ALG statistics:

    Invalid packets received   : 0

    Reply packets received     : 0

    Oversized packets received : 0

  PPTP ALG statistics:

    PPTP Objects Active  : 0
```

Configuring the Stateful Firewall with Application-Level Gateways

To configure the stateful firewall rule:

1. In configuration mode, go to the following hierarchy level:

   user@host# edit services

2. Configure the Stateful Firewall rule.

   [edit services]
   user@host# set stateful-firewall rule *rule*

   In this example, the SFW rule is **rule1 match-direction input-output**.

   [edit services]
   user@host# set stateful-firewall rule rule1 match-direction input-output

3. Configure the rule input conditions for a rule to define the stateful firewall term.

   [edit services]
   user@host# set stateful-firewall rule *rule*

   In this example, the rule input conditions are **rule1 term term1 from applications junos-ftp**, **rule1 term term1 from applications junos-sqlnet**, **rule1 term term1 from applications junos-pptp**, **rule1 term term1 from applications junos-talk-udp**, **rule1 term term1 from applications junos-dns-udp**, and **rule1 term term1 from applications junos-rtsp**

   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-ftp
   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-sqlnet
   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-pptp
   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-talk-udp
   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-dns-udp
   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 from applications junos-rtsp

4. Configure the rule for the stateful firewall term actions.

   [edit services]
   user@host# set stateful-firewall rule *rule*

   In this example, the rule is **rule1 term term1 then accept**.

   [edit services]
   user@host# set stateful-firewall rule rule1 term term1 then accept

5. Verify the configuration.

```
[edit services]
stateful-firewall {
    rule rule1 {
        match-direction input-output;
        term term1 {
            from {
                applications [ junos-ftp junos-sqlnet junos-pptp
junos-talk-udp junos-dns-udp junos-rtsp ];
            }
            then {
                accept;
            }
        }
    }
}
```

Configuring the NAT with ALG

To configure the NAT pool and NAT rule:

1. In configuration mode, go to the following hierarchy level:

   user@host# edit services

2. Configure the NAT pool.

```
[edit services]
user@host# set nat pool pool
```

In this example, the NAT pool is **p1**.

```
[edit services]
user@host# set nat pool p1
```

3. Configure the NAT pool address.

```
[edit services]
user@host# set nat pool p1 address address
```

In this example, the NAT pool address is **20.1.1.10/32**.

```
[edit services]
user@host# set nat pool p1 address 20.1.1.10/32;
```

4. Configure the NAT pool port.

```
[edit services]
user@host# set nat pool p1 port port;
```

In this example, the NAT pool port is **automatic**.

```
[edit services]
user@host# set nat pool p1 port automatic;
```

5. Configure the rule.

```
[edit services]
user@host# set nat rule rule
```

In this example, the rule is **r1**.

```
[edit services]
user@host# set nat rule r1
```

6. Configure the match direction.

```
[edit services]
user@host# set nat rule r1 match-direction match-direction
```

In this example, the match direction is **input**.

```
[edit services]
user@host# set nat rule r1 match-direction input
```

7. Configure the term.

```
[edit services]
user@host# set nat rule r1 term term
```

In this example, the term is **t1**.

```
[edit services]
user@host# set nat rule r1 term t1
```

8. Configure the input conditions for the NAT term.

```
[edit services]
user@host# set nat rule r1 term t1 from from
```

In this example, the input conditions are **applications junos-ftp**, **applications junos-sqlnet**, **applications junos-pptp**, **applications junos-talk-udp**, **applications junos-dns-udp**, **applications junos-rtsp**.

```
[edit services]
user@host# set nat rule r1 term t1 from applications junos-ftp
[edit services]
user@host# set nat rule r1 term t1 from applications junos-sqlnet
[edit services]
user@host# set nat rule r1 term t1 from applications junos-pptp
[edit services]
user@host# set nat rule r1 term t1 from applications junos-talk-udp
[edit services]
user@host# set nat rule r1 term t1 from applications junos-dns-udp
[edit services]
user@host# set nat rule r1 term t1 from applications junos-rtsp
```

9. Configure the NAT term action.

```
[edit services]
user@host# set nat rule r1 term then then
```

In this example, the term action is **translated**.

```
[edit services]
user@host# set nat rule r1 term t1 then translated
```

10. Configure the properties for translated traffic.

```
[edit services]
user@host# set nat rule r1 term then translated translated
```

In this example, the property for the translated traffic is **source-pool p1**.

```
[edit services]
user@host# set nat rule r1 term t1 then translated source-pool p1
```

11. Configure the properties for translated traffic transaction type.

```
[edit services]
user@host# set nat rule r1 term then translated transaction type transaction type
```

In this example, the property for the translated traffic is **source dynamic**.

```
[edit services]
user@host# set nat rule r1 term t1 then translated translation-type source dynamic
```

12. Verify the configuration.

```
[edit services]
user@host# show
services {
        nat {
            pool p1 {
                address 20.1.1.10/32;
                port automatic
            }
            rule r1 {
                match-direction input;
                term t1 {
                    from {
         applications [ junos-ftp junos-sqlnet junos-pptp
```

```
                                                junos-talk-udp junos-dns-udp
junos-rtsp ];
                    }
                    then {
                        translated {
                            source-pool p1;
                            translation-type {
                                source dynamic;
                            }
                        }
                    }
                }
            }
        }
    }
```

CHAPTER 3

Example

Examples: Configuring Application Protocols

The following example shows an application protocol definition describing a special FTP application running on port 78:

```
[edit applications]
application my-ftp-app {
   application-protocol ftp;
   protocol tcp;
   destination-port 78;
   timeout 100; # inactivity timeout for FTP service
}
```

The following example shows a special ICMP protocol (**application-protocol icmp**) of type 8 (ICMP echo):

```
[edit applications]
application icmp-app {
   application-protocol icmp;
   protocol icmp;
   icmp-type icmp-echo;
}
```

The following example shows a possible application set:

```
[edit applications]
application-set basic {
   http;
   ftp;
   telnet;
   nfs;
   icmp;
}
```

The software includes a predefined set of well-known application protocols. The set includes applications for which the TCP and UDP destination ports are already recognized by stateless firewall filters.

CHAPTER 4

Configuration Statements

application

Syntax

```
application application-name {
    application-protocol protocol-name;
    destination-port port-number;
    icmp-code value;
    icmp-type value;
    inactivity-timeout value;
    protocol type;
    rpc-program-number number;
    snmp-command command;
    source-port port-number;
    ttl-threshold number;
    uuid hex-value;
}
```

Hierarchy Level

[edit **applications**],
[edit **applications application-set** *application-set-name*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure properties of an application and whether to include it in an application set.

Options

application-name—Identifier of the application.

The remaining statements are explained separately.

Usage Guidelines

See "Configuring Application Protocol Properties" on page 27.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

application-protocol

Syntax application-protocol *protocol-name*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.
login options introduced in Junos OS Release 7.4.
ip option introduced in Junos OS Release 8.2.

Description Identify the application protocol name. Application protocols are also called application layer gateways (ALGs).

Options ***protocol-name***—Name of the protocol. The following protocols are supported:

- **bootp**
- **dce-rpc**
- **dce-rpc-portmap**
- **dns**
- **exec**
- **ftp**
- **icmp**
- **ip**
- **login**
- **netbios**
- **netshow**
- **rpc**
- **rpc-portmap**
- **rtsp**
- **shell**
- **snmp**
- **sqlnet**
- **tftp**
- **traceroute**

Usage Guidelines See “Configuring an Application Protocol” on page 28.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

application-set

Syntax
```
application-set application-set-name {
    application application-name;
}
```

Hierarchy Level [edit applications]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure one or more applications to include in an application set.

Options ***application-set-name***—Identifier of an application set.

Usage Guidelines See "Configuring Application Sets" on page 36.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

applications

Syntax applications { ... }

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the applications used in services.

Usage Guidelines See Application Properties.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

destination-port

Syntax destination-port *port-value*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number.

Options ***port-value***—Identifier for the port. For a complete list, see "Configuring Source and Destination Ports" on page 32.

Usage Guidelines See "Configuring Source and Destination Ports" on page 32.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

icmp-code

Syntax icmp-code *value*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Internet Control Message Protocol (ICMP) code value.

Options ***value***—The ICMP code value. For a complete list, see "Configuring the ICMP Code and Type" on page 31.

Usage Guidelines See "Configuring the ICMP Code and Type" on page 31.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

icmp-type

Syntax icmp-type *value*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description ICMP packet type value.

Options ***value***—The ICMP type value, such as **echo** or **echo-reply**. For a complete list, see "Configuring the ICMP Code and Type" on page 31.

Usage Guidelines See "Configuring the ICMP Code and Type" on page 31.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

inactivity-timeout

Syntax inactivity-timeout *seconds*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Inactivity timeout period, in seconds.

Options ***seconds***—Length of time the application is inactive before it times out.
Default: 30 seconds

Usage Guidelines See "Configuring the Inactivity Timeout Period" on page 35.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

protocol

Syntax protocol *type*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Networking protocol type or number.

Options ***type***—Networking protocol type. The following text values are supported:

- **ah**
- **egp**
- **esp**
- **gre**
- **icmp**
- **igmp**
- **ipip**
- **ospf**
- **pim**
- **rsvp**
- **tcp**
- **udp**
- **vrrp**

NOTE: IP version 6 (IPv6) is not supported as a network protocol in application definitions.

Usage Guidelines See “Configuring the Network Protocol” on page 29.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

rpc-program-number

Syntax rpc-program-number *number*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Remote procedure call (RPC) or Distributed Computing Environment (DCE) value.

Options ***number***—RPC or DCE program value.
Range: 100,000 through 400,000

Usage Guidelines See “Configuring an RPC Program Number” on page 35.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

snmp-command

Syntax snmp-command *command*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description SNMP command format.

Options ***command***—Supported commands are SNMP **get**, **get-next**, **set**, and **trap**.

Usage Guidelines See “Configuring an SNMP Command for Packet Matching” on page 35.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

source-port

Syntax source-port *port-number*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Source port identifier.

Options ***port-value***—Identifier for the port. For a complete list, see "Configuring Source and Destination Ports" on page 32.

Usage Guidelines See "Configuring Source and Destination Ports" on page 32.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

ttl-threshold

Syntax ttl-threshold *number*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the traceroute time-to-live (TTL) threshold value. This value sets the acceptable level of network penetration for trace routing.

Options ***number***—TTL threshold value.

Usage Guidelines See "Configuring the TTL Threshold" on page 36.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

uuid

Syntax uuid *hex-value*;

Hierarchy Level [edit **applications application** *application-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the Universal Unique Identifier (UUID) for DCE RPC objects.

Options ***hex-value***—Hexadecimal value.

Usage Guidelines See "Configuring a Universal Unique Identifier" on page 36.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

PART 3

Administration

CHAPTER 5

Stateful Firewall Operational Mode Commands

clear services stateful-firewall flows

Syntax clear services stateful-firewall flows
<application-protocol *protocol*>
<destination-port *destination-port*>
<destination-prefix *destination-prefix*>
<interface *interface-name*>
<protocol *protocol*>
<service-set *service-set*>
<source-port *source-port*>
<source-prefix *source-prefix*>

Release Information Command introduced before Junos OS Release 7.4.

Description Clear stateful firewall flows.

Options none—Clear all stateful firewall flows.

destination-port *destination-port*—(Optional) Clear stateful firewall flows for a particular destination port. The range of values is 0 to 65535.

destination-prefix *destination-prefix*—(Optional) Clear stateful firewall flows for a particular destination prefix.

interface *interface-name*—(Optional) Clear stateful firewall flows for a particular interface. On M Series and T Series routers, the ***interface-name*** can be **ms-*fpc/pic/port*** or **rsp*number***. On J Series routers, the ***interface-name*** is **ms-*pim*/0/*port***.

protocol—(Optional) Clear stateful firewall flows for one of the following IP types:

- ***number***—Numeric protocol value from **0** to **255**.
- **ah**—IPsec Authentication Header protocol
- **egp**—An exterior gateway protocol
- **esp**—IPsec Encapsulating Security Payload protocol
- **gre**—A generic routing encapsulation protocol
- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **ipip**—IP-over-IP Encapsulation Protocol
- **ospf**—Open Shortest Path First protocol
- **pim**—Protocol Independent Multicast protocol
- **rsvp**—Resource Reservation Protocol
- **sctp**—Stream Control Protocol
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

service-set *service-set*—(Optional) Clear stateful firewall flows for a particular service set.

source-port *source-port*—(Optional) Clear stateful firewall flows for a particular source port. The range of values is from 0 through 65535.

source-prefix *source-prefix*—(Optional) Clear stateful firewall flows for a particular source prefix.

Required Privilege Level view

Related Documentation

- **show services stateful-firewall flows on page 63**

List of Sample Output **clear services stateful-firewall flows on page 61**

Output Fields Table 6 on page 61 lists the output fields for the **clear services stateful-firewall flows** command. Output fields are listed in the approximate order in which they appear.

Table 6: clear services stateful-firewall flows Output Fields

Field Name	Field Description
Interface	Name of an adaptive services interface.
Service set	Name of the service set from which flows are being cleared.
Conv removed	Number of conversations removed.

clear services stateful-firewall flows

```
user@host> clear services stateful-firewall flows
Interface   Service set                                  Conv removed
ms-0/3/0    svc_set_trust                                       0
ms-0/3/0    svc_set_untrust                                     0
```

clear services stateful-firewall statistics

Syntax
clear services stateful-firewall statistics
<interface *interface-name*>
<service-set *service-set*>

Release Information
Command introduced before Junos OS Release 7.4.

Description
Clear stateful firewall statistics.

Options
none—Clear stateful firewall statistics for all interfaces and all service sets.

interface *interface-name*—(Optional) Clear stateful firewall statistics for the specified interface. On M Series and T Series routers, the ***interface-name*** can be **ms-*fpc/pic/port*** or **rsp*number***. On J Series routers, the ***interface-name*** is **ms-*pim*/0/*port***.

service-set *service-set*—(Optional) Clear stateful firewall statistics for the specified service set.

Required Privilege Level
view

Related Documentation
- **show services stateful-firewall statistics on page 68**

List of Sample Output
clear services stateful-firewall statistics on page 62

Output Fields
When you enter this command, you are provided feedback on the status of your request.

clear services stateful-firewall statistics

```
user@host> clear services stateful-firewall statistics
```

show services stateful-firewall flows

Syntax show services stateful-firewall flows
<brief | extensive | summary | terse>
<application-protocol *protocol*>
<count>
<destination-port *destination-port*>
<destination-prefix *destination-prefix*>
<interface *interface-name*>
<limit *number*>
<protocol *protocol*>
<service-set *service-set*>
<source-port *source-port*>
<source-prefix *source-prefix*>

Release Information Command introduced before Junos OS Release 7.4.
pgcp option introduced in Junos OS Release 8.4.
application-protocol option introduced in Junos OS Release 10.4.

Description Display stateful firewall flow table entries. When the interface is used for softwire processing, the type of softwire concentrator (**DS-LITE** or **6rd**) is shown, and frame counts are provided.

Options none—Display standard information about all stateful firewall flows.

brief | extensive | summary | terse—(Optional) Display the specified level of output.

application-protocol *application-protocol*—(Optional) Display information about one of the following application-level gateway (ALG) protocol types:

- **bootp**—Bootstrap protocol
- **dce-rpc**—Distributed Computing Environment (DCE) remote procedure call (RPC) protocol

NOTE: Use this option to select Microsoft Remote Procedure Call (MSRPC).

- **dce-rpc-portmap**—Distributed Computing Environment (DCE) remote procedure call (RPC) portmap protocol
- **dns**—Domain Name Service protocol
- **exec**—Remote execution protocol
- **ftp**—File Transfer Protocol
- **h323**—H.323 protocol
- **icmp**—Internet Control Message Protocol
- **iiop**—Internet Inter-ORB Protocol

- **ip**—Internet protocol
- **netbios**—NetBIOS protocol
- **netshow**—Netshow protocol
- **pptp** —Point-to-Point Tunneling Protocol
- **realaudio**—RealAudio protocol
- **rpc**—Remote Procedure Call protocol

NOTE: Use this option to select Sun Microsystems Remote Procedure Call protocol (SunRPC).

- **rpc-portmap**—Remote Procedure Call portmap protocol
- **rtsp**—Real-Time Streaming Protocol
- **sip**—Session Initiation Protocol
- **snmp**—Simple Network Management Protocol
- **talk**—Talk protocol
- **tftp**—Trivial File Transfer Protocol
- **traceroute**—Traceroute
- **winframe**—WinFrame

count—(Optional) Display a count of the matching entries.

destination-port *destination-port*—(Optional) Display information for a particular destination port. The range of values is from 0 to 65535.

destination-prefix *destination-prefix*—(Optional) Display information for a particular destination prefix.

interface *interface-name*—(Optional) Display information about a particular interface. On M Series and T Series routers, ***interface-name*** can be **ms-*fpc/pic/port*** or **rsp*number***. On J Series routers, ***interface-name*** is **ms-*pim*/0/*port***.

limit *number*—(Optional) Maximum number of entries to display.

protocol *protocol*—(Optional) Display information about one of the following IP types:

- ***number***—Numeric protocol value from 0 to 255
- **ah**—IPsec Authentication Header protocol
- **egp**—An exterior gateway protocol
- **esp**—IPsec Encapsulating Security Payload protocol
- **gre**—A generic routing encapsulation protocol

- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **ipip**—IP-within-IP Encapsulation Protocol
- **ospf**—Open Shortest Path First protocol
- **pim**—Protocol Independent Multicast protocol
- **rsvp**—Resource Reservation Protocol
- **sctp**—Stream Control Protocol
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

service-set *service-set*—(Optional) Display information for a particular service set.

source-port *source-port*—(Optional) Display information for a particular source port. The range of values is from 0 to 65535.

source-prefix *source-prefix*—(Optional) Display information for a particular source prefix.

Required Privilege Level view

Related Documentation
- **clear services stateful-firewall flows on page 60**

List of Sample Output
show services stateful-firewall flows on page 66
show services stateful-firewall flows (For Softwire Flows) on page 66
show services stateful-firewall flows brief on page 67
show services stateful-firewall flows extensive on page 67
show services stateful-firewall flows count on page 67
show services stateful-firewall flows destination port on page 67
show services stateful-firewall flows source port on page 67
show services stateful-firewall flows (Twice NAT) on page 67

Output Fields Table 7 on page 65 lists the output fields for the **show services stateful-firewall flows** command. Output fields are listed in the approximate order in which they appear.

Table 7: show services stateful-firewall flows Output Fields

Field Name	Field Description
Interface	Name of the interface.
Service set	Name of a service set. Individual empty service sets are not displayed. If no service set has any flows, a flow table header is displayed for each service set.
Flow Count	Number of flows in a session.
Flow or Flow Prot	Protocol used for this flow.

Table 7: show services stateful-firewall flows Output Fields *(continued)*

Field Name	Field Description
Source	Source prefix of the flow in the format ***source-prefix:port***. For ICMP flows, port information is not displayed.
Dest	Destination prefix of the flow. For ICMP flows, port information is not displayed.
State	Status of the flow: • **Drop**—Drop all packets in the flow without response. • **Forward**—Forward the packet in the flow without looking at it. • **Reject**—Drop all packets in the flow with response. • **Watch**—Inspect packets in the flow.
Dir	Direction of the flow: input **(I)** or output **(O)**.
Frm count	Number of frames in the flow.

show services stateful-firewall flows

```
user@host> show services stateful-firewall flows
Interface: ms-1/3/0, Service set: green

Flow
Prot     Source                Dest               State      Dir     Frm count
TCP     10.58.255.178:23   ->    10.59.16.100:4000 Forward    O
TCP      10.58.255.50:33005->   10.58.255.178:23   Forward    I             1
  Source NAT    10.58.255.50:33005->    10.59.16.100:4000
  Destin NAT    10.58.255.178:23   ->         0.0.0.0:4000
```

show services stateful-firewall flows (For Softwire Flows)

When a service set includes softwire processing, the following output format is used for the softwire flows:

```
user@host> show services stateful-firewall flows
Interface: sp-0/1/0, Service set: dslite-svc-set2
Flow                                                 State    Dir       Frm count
TCP      200.200.200.2:80    ->     44.44.44.1:1025  Forward  O          219942
    NAT dest        44.44.44.1:1025    ->       20.20.1.4:1025
    Softwire          2001::2          ->         1001::1
TCP          20.20.1.2:1025  ->  200.200.200.2:80    Forward  I          110244
    NAT source       20.20.1.2:1025    ->      44.44.44.1:1024
    Softwire          2001::2          ->         1001::1
TCP      200.200.200.2:80    ->     44.44.44.1:1024  Forward  O          219140
    NAT dest        44.44.44.1:1024    ->       20.20.1.2:1025
    Softwire          2001::2          ->         1001::1
DS-LITE         2001::2      ->        1001::1       Forward  I          988729
TCP      200.200.200.2:80    ->     44.44.44.1:1026  Forward  O          218906
    NAT dest        44.44.44.1:1026    ->       20.20.1.3:1025
    Softwire          2001::2          ->         1001::1
TCP          20.20.1.3:1025  ->  200.200.200.2:80    Forward  I          110303
    NAT source       20.20.1.3:1025    ->      44.44.44.1:1026
    Softwire          2001::2          ->         1001::1
TCP          20.20.1.4:1025  ->  200.200.200.2:80    Forward  I          110944
    NAT source       20.20.1.4:1025    ->      44.44.44.1:1025
    Softwire          2001::2          ->         1001::1
```

show services stateful-firewall flows brief

The output for the **show services stateful-firewall flows brief** command is identical to that for the **show services stateful-firewall flows** command. For sample output, see **show services stateful-firewall flows**.

show services stateful-firewall flows extensive

```
user@host> show services stateful-firewall flows extensive
Interface: ms-0/3/0, Service set: ss_nat
Flow                                                  State     Dir       Frm
count
TCP           16.1.0.1:2330  ->      16.49.0.1:21        Forward    I
   8
    NAT source        16.1.0.1:2330    ->       16.41.0.1:2330
    NAT dest         16.49.0.1:21      ->       16.99.0.1:21
  Byte count: 455, TCP established, TCP window size: 57344
  TCP acknowledge: 3251737524, TCP tickle enabled, tcp_tickle: 0
  Flow role: Master, Timeout: 720
TCP          16.99.0.1:21    ->      16.41.0.1:2330      Forward    O
   5
    NAT source       16.99.0.1:21      ->       16.49.0.1:21
    NAT dest         16.41.0.1:2330    ->        16.1.0.1:2330
  Byte count: 480, TCP established, TCP window size: 57344
  TCP acknowledge: 463128048, TCP tickle enabled, tcp_tickle: 0
  Flow role: Responder, Timeout: 720
```

show services stateful-firewall flows count

```
user@host> show services stateful-firewall flows count
Interface             Service set                                 Flow Count

ms-1/3/0              green                                              2
```

show services stateful-firewall flows destination port

```
user@router> show services stateful-firewall flows destination-port 21
Interface: ms-0/3/0, Service set: svc_set_trust
Flow
                                                 State    Dir       Frm count
Interface: ms-0/3/0, Service set: svc_set_untrust
Flow                                            State    Dir       Frm count
TCP         10.50.10.2:2143  ->     10.50.20.2:21    Watch    O              0
```

show services stateful-firewall flows source port

```
user@router> show services stateful-firewall flows source-port 2143
Interface: ms-0/3/0, Service set: svc_set_trust
Flow
                                               State    Dir       Frm count
Interface: ms-0/3/0, Service set: svc_set_untrust
Flow                                            State    Dir       Frm count
TCP         10.50.10.2:2143  ->     10.50.20.2:21    Watch    O              0
```

show services stateful-firewall flows (Twice NAT)

```
user@router> show services stateful-firewall flows
Flow                                           State    Dir       Frm count
UDP          40.0.0.8:23439 ->     80.0.0.1:16485   Watch    I            20
    NAT source        40.0.0.8:23439   ->     172.16.1.10:1028
    NAT dest          80.0.0,1:16485   ->     192.16.1.10:22415
UDP      192.16.1.10:22415  ->  172.16.1.10:1028    Watch    O            20
    NAT source     192.16.1.10:22415   ->        80.0.0.1:16485
    NAT dest       172.16.1.10:1028    ->        40.0.0.8:23439
```

show services stateful-firewall statistics

Syntax

show services stateful-firewall statistics
<application-protocol *protocol*>
<brief | detail | extensive | summary>
<interface *interface-name*>
<service-set *service-set*>

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display stateful firewall statistics.

Options

none—Display standard information about all stateful firewall statistics.

brief | detail | extensive | summary—(Optional) Display the specified level of output.

interface *interface-name*—(Optional) Display information about a particular interface. On M Series and T Series routers, the ***interface-name*** can be **ms-*fpc/pic/port*** or **rsp*number***. On J Series routers, the ***interface-name*** is **ms-*pim*/0/*port***.

service-set *service-set*—(Optional) Display information about a particular service set.

Required Privilege Level

view

Related Documentation

- **clear services stateful-firewall statistics on page 62**

List of Sample Output

show services stateful-firewall statistics extensive on page 71

Output Fields

Table 8 on page 68 lists the output fields for the **show services stateful-firewall statistics** command. Output fields are listed in the approximate order in which they appear.

Table 8: show services stateful-firewall statistics Output Fields

Field Name	Field Description
Interface	Name of an adaptive services interface.
Service set	Name of a service set.
New flows	Rule match counters for new flows: • **Accept**—New flows accepted. • **Discard**—New flows discarded. • **Reject**—New flows rejected.
Existing flows	Rule match counters for existing flows: • **Accept**—Match existing forward or watch flow. • **Discard**—Match existing discard flow. • **Reject**—Match existing reject flow.

Table 8: show services stateful-firewall statistics Output Fields *(continued)*

Field Name	Field Description
Drops	Drop counters: • **TCP SYN defense**—Packets dropped by SYN defender. • **NAT ports exhausted**—Hide mode. The router has no available Network Address Translation (NAT) ports for a given address or pool.
Errors	Total errors, categorized by protocol: • **IP**—Total IP version 4 errors. • **TCP**—Total Transmission Control Protocol (TCP) errors. • **UDP**—Total User Datagram Protocol (UDP) errors. • **ICMP**—Total Internet Control Message Protocol (ICMP) errors. • **Non-IP**—Total non-IPv4 errors.
IP Errors	IPv4 errors: • **IP packet length inconsistencies**—IP packet length does not match the Layer 2 reported length. • **Minimum IP header length check failures**—Minimum IP header length is 20 bytes. The received packet contains less than 20 bytes. • **Reassembled packet exceeds maximum IP length**—After fragment reassembly, the reassembled IP packet length exceeds 65,535. • **Illegal source address 0**—Source address is not a valid address. Invalid addresses are, loopback, broadcast, multicast, and reserved addresses. Source address **0**, however, is allowed to support BOOTP and the destination address **0xffffffff**. • **Illegal destination address 0**—Destination address is not a valid address. The address is reserved. • **TTL zero errors**—Received packet had a time-to-live (TTL) value of 0. • **IP protocol number 0 or 255**—IP protocol is 0 or 255. • **Land attack**—IP source address is the same as the destination address. • **Smurf attack**—Echo request is sent to a directed broadcast address. • **Non-IP packets**—Packet did not conform to the IP standard. • **IP option**—Packet dropped because of a nonallowed IP option. • **Non-IPv4 packets**—Packet was not IPv4. (Only IPv4 is supported.) • **Bad checksum**—Packet had an invalid IP checksum. • **Illegal IP fragment length**—Illegal fragment length. All fragments (other than the last fragment) must have a length that is a multiple of 8 bytes. • **IP fragment overlap**—Fragments have overlapping fragment offsets. • **IP fragment reassembly timeout**—Some of the fragments for an IP packet were not received in time, and the reassembly handler dropped partial fragments.

Table 8: show services stateful-firewall statistics Output Fields *(continued)*

Field Name	Field Description
TCP Errors	TCP protocol errors: • **TCP header length inconsistencies**—Minimum TCP header length is 20 bytes, and the IP packet received does not contain at least 20 bytes. • **Source or destination port number is zero**—TCP source or destination port is zero. • **Illegal sequence number, flags combination**—Dropped because of TCP errors, such as an illegal sequence number, which causes an illogical combination of flags to be set. • **SYN attack (multiple SYN messages seen for the same flow)**—Multiple SYN packets received for the same flow are treated as a SYN attack. The packets might be retransmitted SYN packets and therefore valid, but a large number is cause for concern. • **First packet not SYN**—First packets for a connection are not SYN packets. These packets might originate from previous connections or from someone performing an ACK/FIN scan. • **TCP port scan (Handshake, RST seen from server for SYN)**—In the case of a SYN defender, if an RST (reset) packet is received instead of a SYN/ACK message, someone is probably trying to scan the server. This behavior can result in false alarms if the RST packet is not combined with an intrusion detection service (IDS). • **Bad SYN cookie response**—SYN cookie generates a SYN/ACK message for all incoming SYN packets. If the ACK received for the SYN/ACK message does not match, this counter is incremented.
UDP Errors	UDP protocol errors: • **IP data length less than minimum UDP header length (8 bytes)**—Minimum UDP header length is 8 bytes. The received IP packets contain less than 8 bytes. • **Source or destination port is zero**—UDP source or destination port is **0**. • **UDP port scan (ICMP error seen for UDP flow)**—ICMP error is received for a UDP flow. This could be a genuine UDP flow, but it is counted as an error.
ICMP Errors	ICMP protocol errors: • **IP data length less than minimum ICMP header length (8 bytes)**—ICMP header length is 8 bytes. This counter is incremented when received IP packets contain less than 8 bytes. • **ICMP error length inconsistencies**—Minimum length of an ICMP error packet is 48 bytes, and the maximum length is 576 bytes. This counter is incremented when the received ICMP error falls outside this range. • **Ping duplicate sequence number**—Received ping packet has a duplicate sequence number. • **Ping mismatched sequence number**—Received ping packet has a mismatched sequence number.

show services stateful-firewall statistics extensive

```
user@host> show services stateful-firewall statistics extensive
Interface: ms-1/3/0
  Service set: interface-svc-set
    New flows:
      Accept: 907, Discard: 0, Reject: 0
    Existing flows:
      Accept: 3535, Discard: 0, Reject: 0
    Drops:
      IP option: 0, TCP SYN defense: 0
      NAT ports exhausted: 0
    Errors:
      IP: 0, TCP: 0
      UDP: 0, ICMP: 0
      Non-IP packets: 0, ALG: 0
    IP errors:
      IP packet length inconsistencies: 0
      Minimum IP header length check failures: 0
      Reassembled packet exceeds maximum IP length: 0
      Illegal source address: 0
      Illegal destination address: 0
      TTL zero errors: 0, IP protocol number 0 or 255: 0
      Land attack: 0, Smurf attack: 0
      Non IP packets: 0, IP option: 0
      Non-IPv4 packets: 0, Bad checksum: 0
      Illegal IP fragment length: 0
      IP fragment overlap: 0
      IP fragment reassembly timeout: 0
TCP errors:
      TCP header length inconsistencies: 0
      Source or destination port number is zero: 0
      Illegal sequence number, flags combination: 0
      SYN attack (multiple SYNs seen for the same flow): 0
      First packet not SYN: 0
      TCP port scan (Handshake, RST seen from server for SYN): 0
      Bad SYN cookie response: 0
    UDP errors:
      IP data length less than minimum UDP header length (8 bytes): 0
      Source or destination port is zero: 0
      UDP port scan (ICMP error seen for UDP flow): 0
    ICMP errors:
      IP data length less than minimum ICMP header length (8 bytes): 0
      ICMP error length inconsistencies: 0
      Ping duplicate sequence number: 0
      Ping mismatched sequence number: 0
    ALG drops:
      BOOTP: 0, DCE-RPC: 0, DCE-RPC portmap: 0
      DNS: 0, Exec: 0, FTP: 0
      ICMP: 0
      Login: 0, Netbios: 0, Netshow: 0
      RPC: 0, RPC portmap: 0
      RTSP: 0, Shell: 0
      SNMP: 0, Sqlnet: 0, TFTP: 0
      Traceroute: 0
```

PART 4

Troubleshooting

CHAPTER 6

Knowledge Base

PART 5

Index

Index