



Junos[®] OS

Application Aware Services Interfaces Feature Guide for Routing Devices

Release
15.1



Modified: 2016-06-28

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Junos[®] OS Application Aware Services Interfaces Feature Guide for Routing Devices

15.1

Copyright © 2016, Juniper Networks, Inc.
All rights reserved.

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

	About the Documentation	xi
	Documentation and Release Notes	xi
	Supported Platforms	xi
	Using the Examples in This Manual	xi
	Merging a Full Example	xii
	Merging a Snippet	xii
	Documentation Conventions	xiii
	Documentation Feedback	xv
	Requesting Technical Support	xv
	Self-Help Online Tools and Resources	xv
	Opening a Case with JTAC	xvi
Chapter 1	Configuring Stateless, Rule-Based Services Using Application-Aware Access Lists	17
	AACL Overview	17
	Best-Effort Application Identification of DPI-Serviced Flows	18
	Features That Support Application-Level Filtering	19
	Best-Effort Application Determination	19
	APPID, AACL, and L-PDF Processing in Preconvergence Scenarios	19
	Prior to a Final or Best-Effort Application Identification	19
	Upon Best-Effort Application Identification	20
	While Application Identification Is on a Best-Effort Basis	20
	If a Flow Ends Before an Application Identification Is Made	20
	If a Flow Ends While Application Identification on a Best-Effort Basis	20
	Configuring AACL Rules	21
	Configuring Match Direction for AACL Rules	22
	Configuring Match Conditions in AACL Rules	22
	Configuring Actions in AACL Rules	24
	Logging AACL Flows Based on Application	25
	Example: Configuring AACL Rules	26
	Configuring AACL Rule Sets	26
	Configuring Logging of AACL Flows	27
Chapter 2	Grouping Applications Together Using APPID	29
	APPID Overview	29
	Best-Effort Application Identification of DPI-Serviced Flows	31
	Features That Support Application-Level Filtering	32
	Best-Effort Application Determination	32

	APPID, ACL, and L-PDF Processing in Preconvergence Scenarios	32
	Prior to a Final or Best-Effort Application Identification	32
	Upon Best-Effort Application Identification	33
	While Application Identification Is on a Best-Effort Basis	33
	If a Flow Ends Before an Application Identification Is Made	33
	If a Flow Ends While Application Identification on a Best-Effort Basis	33
	Defining an Application Identification	34
	Configuring APPID Rules	36
	Using Stateful Firewall Rules to Identify Data Sessions	37
	Configuring Application Profiles	39
	Configuring Application Groups	40
	Application Identification for Nested Applications	41
	Disabling Application Identification for Nested Applications	42
	Configuring Global APPID Properties	43
	Configuring APPID Support for Heuristics	44
	Configuring APPID Support for Unidirectional Traffic	45
	Configuring Automatic Download of Application Package Updates	46
	Tracing APPID Operations	46
	Configuring the APPID Log Filename	47
	Configuring the Number and Size of APPID Log Files	47
	Configuring Access to the Log File	47
	Configuring a Regular Expression for Lines to Be Logged	48
	Configuring the Tracing Flags	48
	Examples: Configuring Application Identification Properties	48
Chapter 3	Detecting Suspicious and Anomalous Network Traffic Using IDP	51
	IDP Overview	51
	Best-Effort Application Identification of DPI-Serviced Flows	53
	Features That Support Application-Level Filtering	53
	Best-Effort Application Determination	54
	APPID, ACL, and L-PDF Processing in Preconvergence Scenarios	54
	Prior to a Final or Best-Effort Application Identification	54
	Upon Best-Effort Application Identification	55
	While Application Identification Is on a Best-Effort Basis	55
	If a Flow Ends Before an Application Identification Is Made	55
	If a Flow Ends While Application Identification on a Best-Effort Basis	55
	Configuring Multiple IDP Detectors	56
Chapter 4	Collecting Statistics and Tracking Data Using L-PDF	57
	L-PDF Overview	57
	Best-Effort Application Identification of DPI-Serviced Flows	59
	Features That Support Application-Level Filtering	59
	Best-Effort Application Determination	60
	APPID, ACL, and L-PDF Processing in Preconvergence Scenarios	60
	Prior to a Final or Best-Effort Application Identification	60
	Upon Best-Effort Application Identification	61
	While Application Identification Is on a Best-Effort Basis	61
	If a Flow Ends Before an Application Identification Is Made	61

	If a Flow Ends While Application Identification on a Best-Effort Basis	61
	Configuring Statistics Profiles	62
	Configuring an L-PDF Statistics Profile	63
	Configuring an ACL Statistics Profile	64
	Applying L-PDF Profiles to Service Sets	65
	Tracing L-PDF Operations	67
Chapter 5	Configuration Statements	69
	[edit services aacl] Hierarchy List	71
	[edit services application-identification] Hierarchy Level	72
	aac1-fields	75
	aac1-statistics-profile	76
	address	77
	application (Defining)	78
	application (Including in Rule)	79
	application-aware-access-list-fields	80
	application-group	81
	application-group-any	81
	application-groups (Services ACL)	82
	application-groups (Services Application Identification)	82
	application-system-cache-timeout	83
	application-unknown	83
	applications (Services ACL)	83
	applications (Services Application Identification)	84
	automatic	84
	bypass-traffic-on-exceeding-flow-limits	85
	chain-order	85
	context	86
	destination (Services)	86
	destination-address	87
	destination-address-range	87
	destination-prefix-list	88
	direction	88
	disable (APPID Application)	89
	disable (APPID Application Group)	89
	disable (APPID Port Mapping)	89
	disable-global-timeout-override	90
	download	90
	enable-asymmetric-traffic-processing	91
	enable-heuristics	91
	file	92
	from	93
	idle-timeout	94
	ignore-errors	94
	index (Applications)	95
	index (Nested Applications)	95
	inactivity-non-tcp-timeout	96
	inactivity-tcp-timeout	96

ip	97
local-policy-decision-function	98
log (aACL)	99
match-direction	99
max-checked-bytes	100
maximum-transactions	100
member	101
min-checked-bytes	101
nested-application	102
nested-applications	103
nested-application-settings	103
nested-application-unknown	104
no-application-identification	104
no-application-system-cache	105
no-clear-application-system-cache	105
no-nested-application	106
no-protocol-method	106
no-signature-based	107
order (Services Application Identification)	107
pattern	108
policy-decision-statistics-profile	109
port-mapping	110
port-range	110
profile	111
protocol	111
rule (AACL Rule Set)	112
rule (Application Identification)	113
rule (Including in Rule Set)	114
rule-set (Services AACL)	114
rule-set (Services Application Identification)	115
service-set-options	116
statistics (System Services)	117
support-uni-directional-traffic	117
service-set (Services)	118
services (AACL)	120
services (Application Identification)	120
session-timeout (Application Identification)	121
session-timeout (Interfaces)	121
signature	122
signature-method-all-ports	122
source	123
source-address (AACL)	123
source-address-range	124
source-prefix-list (Services AACL)	124
source-prefix-list (Services IDS)	125
term	126
then	127
traceoptions (Application Identification)	129
traceoptions (Services Local Policy Decision Function)	131

	type	132
	type-of-service	133
	url	133
Chapter 6	Operational Commands	135
	clear services application-aware-access-list statistics	136
	clear services application-identification application-system-cache	137
	clear services application-identification counter	138
	clear services flows	139
	clear services flows ip-action	142
	clear services local-policy-decision-function statistics	143
	request services application-identification application	144
	request services application-identification download	146
	request services application-identification download status	147
	request services application-identification group	148
	request services application-identification install	150
	request services application-identification install status	151
	show services application-aware-access-list flows	152
	show services application-identification application-system-cache	155
	show services application-identification counter	157
	show services application-identification group	160
	show services application-aware-access-list statistics	162
	show services application-identification application	164
	show services application-identification version	167
	show services flows	168
	show services local-policy-decision-function flows	175
	show services local-policy-decision-function statistics	177
	show services sessions	179
Chapter 7	Index	187
	Index	189

List of Tables

Chapter 6

About the Documentation	xi
Table 1: Notice Icons	xiii
Table 2: Text and Syntax Conventions	xiv
Operational Commands	135
Table 3: clear services flows Output Fields	141
Table 4: show services application-aware-access-list flows Output Fields	152
Table 5: show application-identification application-system-cache Output Fields	155
Table 6: show services application-identification counter Output Fields	157
Table 7: show services application-identification group Output Fields	160
Table 8: show services application-aware-access-list statistics Output Fields	162
Table 9: show services application-identification application Output Fields ..	164
Table 10: show services flows Output Fields	170
Table 11: show services local-policy-decision-function flows Output Fields ..	175
Table 12: show services local-policy-decision-function statistics Output Fields	177
Table 13: show services sessions Output Fields	182

About the Documentation

- Documentation and Release Notes on page xi
- Supported Platforms on page xi
- Using the Examples in This Manual on page xi
- Documentation Conventions on page xiii
- Documentation Feedback on page xv
- Requesting Technical Support on page xv

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- M Series
- MX Series
- T Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

Documentation Conventions

Table 1 on page xiii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xiv defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces or emphasizes important new terms. Identifies guide names. Identifies RFC and Internet draft titles. 	<ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS CLI User Guide</i> RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE.
< > (angle brackets)	Encloses optional keywords or variables.	stub <default-metric metric>;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast <i>(string1 string2 string3)</i>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Encloses a variable for which you can substitute one or more values.	community name members [community-ids]
Indentation and braces ({ })	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

GUI Conventions

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"> In the Logical Interfaces box, select All Interfaces. To cancel the configuration, click Cancel.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select Protocols>Ospf .

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.

CHAPTER 1

Configuring Stateless, Rule-Based Services Using Application-Aware Access Lists

- [AACL Overview on page 17](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 18](#)
- [Configuring AACL Rules on page 21](#)
- [Example: Configuring AACL Rules on page 26](#)
- [Configuring AACL Rule Sets on page 26](#)
- [Configuring Logging of AACL Flows on page 27](#)

AACL Overview



NOTE: Starting with Junos OS Release 12.1, all interface-style services are supported for dynamic Point-to-Point Protocol over Ethernet (PPPoE) subscribers on all MX Series routers with modular Modular Port Concentrators (MPCs).

The application-aware access list (AACL) service adds support for a new service that uses application names and groups as matching criteria for filtering traffic. AACL is a stateless, rules-based service that must be combined with application identification to enable policies to be applied to flows based on application and application group membership in addition to traditional packet matching rules. It is supported on MX Series routers equipped with Multiservices DPCs and on M120 or M320 routers equipped with Multiservices 400 PICs. Starting with Junos OS Release 11.3, AACL is supported on T320, T640, and T1600 routers also.

AACL is configured in a similar way to other rules-based services such as Network Address Translation (NAT), class of service (CoS), and stateful firewall. To configure AACL, include rule specifications for match criteria and actions at the **[edit services aacl]** hierarchy level. You can chain AACL rules along with other service rules by including them in a service-set definition at the **[edit services service-set]** hierarchy level, as previously documented.

There is one pair of related operational commands, **show/clear application-aware-access-list statistics**.

For more information on the CLI configuration, see the *Application Aware Services Interfaces Feature Guide for Routing Devices*. For more information on the operational command, see the [CLI Explorer](#).



NOTE: Because the Junos OS extension-provider package framework lacks aggressive constraint checks, you should not set the `policy-db-size` statement at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level to a high value. For Junos Application Aware (previously known as dynamic application awareness) configurations, the recommended values for the extension-provider options at this hierarchy level are as follows:

- `control-cores = 1`
- `data-cores = 7`
- `object-cache-size = 1280` (for Multiservices 400 PIC and Multiservices DPC)
- `policy-db-size = 200`
- Include these package values: `jservices-idp`, `jservices-appid`, `jservices-llpdf`, `jservices-aacl`

For more information about this configuration, see the following topics in the *SDK Applications Configuration Guide and Command Reference*:

- *Configuring Control and Data Cores*
- *Configuring Memory Settings*
- *Configuring Packages on the PIC*

**Related
Documentation**

- [Configuring AACL Rules on page 21](#)
- [Configuring AACL Rule Sets on page 26](#)
- [Configuring Logging of AACL Flows on page 27](#)
- [Example: Configuring AACL Rules on page 26](#)

Best-Effort Application Identification of DPI-Serviced Flows

This topic describes the following information:

- [Features That Support Application-Level Filtering on page 19](#)
- [Best-Effort Application Determination on page 19](#)
- [APPID, AACL, and L-PDF Processing in Preconvergence Scenarios on page 19](#)

Features That Support Application-Level Filtering

On MX Series routers equipped with Multiservices DPCs and M120 or M320 routers equipped with Multiservices 400 PICs, Intrusion Detection and Prevention (IDP) is accomplished by Deep Packet Inspection (DPI) of TCP, UDP, and ICMP flows. The application identification (APPID) feature defines applications as members of application groups in TCP/UDP/ICMP traffic. IDP depends on APPID for identification and detection of some Layer 7 applications.

The application-aware access list (AACL) service uses application names and groups as matching criteria for filtering traffic. The service defines the applications and application groups for which statistics are collected for a specific user or interface.

The local policy decision function (L-PDF) enables you to configure properties for statistics output. L-PDF supports a process that regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

Best-Effort Application Determination

Typically, APPID conclusively determines the Layer 7 application associated with a given DPI-serviced flow. In these cases, the application identification is final. Occasionally, APPID is only able to make an initial, inconclusive determination of the Layer 7 application associated with a given flow. This is referred to as a "best-effort" application identification. In such cases, the APPID process continues processing packets on that flow and might subsequently make a conclusive determination of the application associated with that flow. In some cases of best-effort application identification, the flow ends before a final application determination can be made.

APPID, AACL, and L-PDF Processing in Preconvergence Scenarios

The following sections describe APPID, AACL, and L-PDF processing in various stages of application identification for a DPI-serviced flow of TCP/UDP/ICMP traffic.

- [Prior to a Final or Best-Effort Application Identification on page 19](#)
- [Upon Best-Effort Application Identification on page 20](#)
- [While Application Identification Is on a Best-Effort Basis on page 20](#)
- [If a Flow Ends Before an Application Identification Is Made on page 20](#)
- [If a Flow Ends While Application Identification on a Best-Effort Basis on page 20](#)

Prior to a Final or Best-Effort Application Identification

During the time that APPID has not yet made either a final or best-effort determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has not yet made either a final or best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has not yet made either a final or best-effort determination of the associated application.

Upon Best-Effort Application Identification

When a best-effort application determination is made, AACL does not apply any AACL term actions configured for that flow. There are a number of reasons for this, one being that the action itself (such as "discard") could make a final application determination impossible. Instead, AACL or L-PDF tracks the flow and accepts all packets for that flow until a final determination is made, at which time the normal AACL or L-PDFL actions are fully applied to the flow.

While Application Identification Is on a Best-Effort Basis

During the time that APPID identification of the application associated with a given flow is on a best-effort basis, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has only made a best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has only made a best-effort determination of the associated application.

If a Flow Ends Before an Application Identification Is Made

If a flow ends before APPID has made either a final or a best-effort application identification, AACL or L-PDF uses the "unknown" application ID as a final determination and performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** AACL term action is configured for the "application-group-any" application, then the statistics for that flow will be collected and aggregated against the count bucket type, and reported as such.

If a Flow Ends While Application Identification on a Best-Effort Basis

If a flow ends while the application identification is on a best-effort basis, AACL or L-PDF uses that best-effort determination as a final determination. AACL or L-PDF performs

any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** AACL term action is configured for that Layer 7 application, then the statistics for the flow will be collected and aggregated against the AACL or L-PDF statistics. However, in the case of nested applications, AACL and L-PDF will not consider the best-effort determination as final and the nested application will be reported as an unknown application.

Related Documentation

- [Configuring AACL Rules on page 21](#)
- [Configuring Statistics Profiles on page 62](#)
- [aACL-fields on page 75](#)
- [aACL-statistics-profile on page 76](#)
- [rule on page 112](#)
- [services on page 120](#)
- [term on page 126](#)
- [then on page 127](#)

Configuring AACL Rules

To configure an AACL rule, include the **rule** *rule-name* statement at the **[edit services aACL]** hierarchy level:

```
rule rule-name {
  match-direction (input | output | input-output);
  term term-name {
    from {
      application-group-any;
      application-groups [ application-group-names ];
      applications [ application-names ];
      destination-address address <any-unicast>;
      destination-address-range low minimum-value high maximum-value;
      destination-prefix-list list-name;
      nested-applications [ nested-application-names ];
      nested-application-unknown;
      source-address address <any-unicast>;
      source-address-range low minimum-value high maximum-value;
      source-prefix-list list-name;
    }
    then {
      (accept | discard);
      count (application | application-group | application-group-any | nested-application
        | none);
      forwarding-class class-name;
      policer policer-name;
    }
  }
}
```

Each AACL rule consists of a set of terms, similar to a filter configured at the **[edit firewall]** hierarchy level. A term consists of the following:

- **from** statement—Specifies the match conditions and applications that are included and excluded.
- **then** statement—Specifies the actions and action modifiers to be performed by the router software.

The following sections explain how to configure the components of ACL rules:

- [Configuring Match Direction for ACL Rules on page 22](#)
- [Configuring Match Conditions in ACL Rules on page 22](#)
- [Configuring Actions in ACL Rules on page 24](#)
- [Logging ACL Flows Based on Application on page 25](#)

Configuring Match Direction for ACL Rules

Each rule must include a **match-direction** statement that specifies the direction in which the rule match is applied. To configure where the match is applied, include the **match-direction** statement at the **[edit services acl rule *rule-name*]** hierarchy level:

```
match-direction (input | output | input-output);
```

If you configure **match-direction input-output**, bidirectional rule creation is allowed.

The match direction is used with respect to the traffic flow through the services PIC or DPC. When a packet is sent to the PIC or DPC, direction information is carried along with it.

With an interface service set, packet direction is determined by whether a packet is entering or leaving the interface on which the service set is applied.

With a next-hop service set, packet direction is determined by the interface used to route the packet to the services PIC or DPC. If the inside interface is used to route the packet, the packet direction is input. If the outside interface is used to direct the packet to the PIC or DPC, the packet direction is output. For more information on inside and outside interfaces, see *Configuring Service Sets to be Applied to Services Interfaces*.

On the PIC or DPC, a flow lookup is performed. If no flow is found, rule processing is performed. All rules in the service set are considered. During rule processing, the packet direction is compared against rule directions. Only rules with direction information that matches the packet direction are considered.

Configuring Match Conditions in ACL Rules

To configure ACL match conditions, include the **from** statement at the **[edit services acl rule *rule-name* term *term-name*]** hierarchy level:

```
from {  
  application-group-any;  
  application-groups [ application-group-names ];  
  applications [ application-names ];  
  destination-address address <any-unicast>;  
  destination-address-range low minimum-value high maximum-value;  
  destination-prefix-list list-name;
```

```

nested-applications [ nested-application-names ];
nested-application-unknown
source-address address <any-unicast>;
source-address-range low minimum-value high maximum-value;
source-prefix-list list-name;
}

```

IPv4 and IPv6 source and destination addresses are supported. You can use either the source address or the destination address as a match condition, in the same way that you configure a firewall filter; for more information, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Alternatively, you can specify a list of source or destination prefixes by configuring the **prefix-list** statement at the **[edit policy-options]** hierarchy level and then including either the **destination-prefix-list** or the **source-prefix-list** statement in the AACL rule. For an example, see “[Example: Configuring AACL Rules](#)” on page 26.

If you omit the **from** term, the AACL rule accepts all traffic and the default protocol handlers take effect:

- User Datagram Protocol (UDP), Transmission Control Protocol (TCP), and Internet Control Message Protocol (ICMP) create a bidirectional flow with a predicted reverse flow.
- IP creates a unidirectional flow.

You can also include application and application group definitions you have configured at the **[edit services application-identification]** hierarchy level; for more information, see the topics in *Application Identification*.

- To apply one or more specific application protocol definitions, include the **applications** statement at the **[edit services aacl rule rule-name term term-name from]** hierarchy level.
- To apply one or more sets of application group definitions you have defined, include the **application-groups** statement at the **[edit services aacl rule rule-name term term-name from]** hierarchy level.



NOTE: If you include one of the statements that specifies application protocols, the router derives port and protocol information from the corresponding configuration at the **[edit services application-identification]** hierarchy level; you cannot specify these properties as match conditions.

- To consider any application group defined in the database as a match, include the **application-group-any** statement at the **[edit services aacl rule rule-name term term-name from]** hierarchy level.
- To consider any nested application defined in the database a match, include the **nested-applications** statement at the **[edit services aacl rule rule-name term term-name from]** hierarchy level. Nested applications are protocols that run on a parent application. For example, if the Facebook application runs on the parent application `junos:http`, the nested application will be `junos:http:facebook`.

Configuring Actions in ACL Rules

To configure ACL actions, include the **then** statement at the **[edit services aacl rule rule-name term term-name]** hierarchy level:

```
then {  
  (accept | discard);  
  (count (application | application-group | application-group-any | nested-application |  
    none) | forwarding-class class-name);  
}
```

You must include one of the following actions:

- **accept**—The packet is accepted and sent on to its destination.
- **discard**—The packet is not accepted and is not processed further.

When you select **accept** as the action, you can optionally configure one or both of the following action modifiers. No action modifiers are allowed with the **discard** action.

- **count (application | application-group | application-group-any | nested-application | none)**—For all accepted packets that match the rules, record a packet count using ACL statistics practices. You can specify one of the following options; there is no default setting:
 - **application**—Count the application that matched in the **from** clause.
 - **application-group**—Count the application group that matched in the **from** clause.
 - **application-group-any**—Count all application groups that match **from** **application-group-any** under the **any** group name.
 - **nested-application**—Count all nested applications that matched in the **from** clause.
 - **none**—Same as not specifying **count** as an action.



NOTE:

- When a session closes before APPID has identified nested applications, the session is treated as a best-effort session and ACL does not get the nested application information. In such cases, nested applications will be reported as unknown applications.
- During the time that the application identification (APPID) feature has not yet made a final determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection. For more information, see [“Best-Effort Application Identification of DPI-Serviced Flows” on page 18](#).

-
- **forwarding-class class-name**—Specify the packets’ forwarding-class name.

You can optionally include a **policer** that has been specified at the **[edit firewall]** hierarchy level. Only the bit-rate and burst-size properties specified for the policer are applied in the ACL rule set. The only action application when a policer is configured is **discard**. For

more information on policer definitions, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Logging AACL Flows Based on Application

You can now log AACL flows based on application. You can select a specific application or request information on unknown applications.

You can now configure AACL rules to match unknown applications. All existing actions that can apply to recognized applications can also apply to unknown applications. You can use the following statements at the **[edit services aacl rule rule-name term term-name from]** hierarchy level:

- application-group-any
- application-groups
- application-unknown
- applications
- nested-application-unknown
- nested-applications

The addition of matching “application unknown” enables the specific logging of the input flows associated with applications that cannot be identified. Because logging is triggered by an input event, you must specify **match-direction** as **input-output** or **input**.

To configure logging of flows for AACL, include the **match-direction input** or **match-direction input-output** statement at the **[edit services aacl rule rule-name]** hierarchy level, include an **applications** or **application-unknown** statement at the **[edit services aacl rule rule-name term term-name from]** hierarchy level, and include only one **log** statement at the **[edit services aacl rule rule-name term term-name then]** hierarchy level. The log statements can include any of the following options:

- session-start
- session-end
- session-start-end-no-stats
- session-start-interim-end
- session-interim-end
- session-end

Related Documentation

- [AACL Overview on page 17](#)
- [Configuring AACL Rule Sets on page 26](#)
- [Configuring Logging of AACL Flows on page 27](#)
- [Example: Configuring AACL Rules on page 26](#)

Example: Configuring AACL Rules

The following example shows an AACL configuration containing a rule with three terms using a variety of match conditions and actions:

```
[edit services aacl]
rule aacl-test {
  match-direction input;
  term term1 {
    from {
      source-address 10.0.1.1
      application test1;
    }
    then {
      accept;
    }
  }
  term term2 {
    from {
      source-address {
        any-unicast;
      }
      application test1;
    }
    then {
      discard;
    }
  }
  term term3 {
    from {
      source-address {
        any-unicast;
      }
      application test1 test2;
    }
    then {
      accept;
      count application;
    }
  }
}
```

- Related Documentation**
- [AACL Overview on page 17](#)
 - [Configuring AACL Rules on page 21](#)

Configuring AACL Rule Sets

The **rule-set** statement defines a collection of AACL rules that determine what actions the router software performs on packets in the data stream. You define each rule by specifying a rule name and configuring terms. Then, you specify the order of the rules by

including the **rule-set** statement at the **[edit services aacl]** hierarchy level with a **rule** statement for each rule:

```
rule-set rule-set-name {
  rule rule-name;
}
```

The router software processes the rules in the order in which you specify them in the configuration. If a term in a rule matches the packet, the router performs the corresponding action and the rule processing stops. If no term in a rule matches the packet, processing continues to the next rule in the rule set. If none of the rules matches the packet, the packet is dropped by default.

Related Documentation

- [AACL Overview on page 17](#)
- [Configuring AACL Rules on page 21](#)
- [Configuring Logging of AACL Flows on page 27](#)
- [Example: Configuring AACL Rules on page 26](#)

Configuring Logging of AACL Flows

You can configure logging of AACL flows for a given application or for all unknown applications using AACL rules. You must set **match-direction** to **input** or **input-output** for logging to occur.

1. Create a rule and term.

```
user@host# edit services aacl rule rule-name term term-name
```

2. Specify selection of an application.

```
[edit services aacl rule rule-name term term-name]
user@host# set from applications application-name]
```

OR

Specify selection of all unknown applications.

```
[edit services aacl rule <variable>rule-name</variable> term
<variable>term-name</variable>]
set from application-unknown
```

3. In the **then** statement, specify logging of input flow.

```
[edit services aacl rule rule-name term term-name]
user@host# set then log input-flows]
```

Example—Configuration of Logging of Input Flows for Unknown Applications

```
[edit services aacl rule aac_rule5]
match-direction input-output;
term t0 {
  from {
    application-unknown;
  }
  then {
    count application;
```

```
        log input-flow;
        accept;
    }
}
```

Example—Setup of a Specific Log File

The following example shows how to direct the aacL flow log to a file other than the default syslog file on the Routing Engine file system.

```
[edit system syslog]
file aacL_log {
    external any;
    match aacL-flow-log;
}
```

Related Documentation

- [AACL Overview on page 17](#)
- [Configuring AACL Rules on page 21](#)
- [Configuring AACL Rule Sets on page 26](#)
- [Example: Configuring AACL Rules on page 26](#)

CHAPTER 2

Grouping Applications Together Using APPID

- [APPID Overview on page 29](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 31](#)
- [Defining an Application Identification on page 34](#)
- [Configuring APPID Rules on page 36](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 37](#)
- [Configuring Application Profiles on page 39](#)
- [Configuring Application Groups on page 40](#)
- [Application Identification for Nested Applications on page 41](#)
- [Disabling Application Identification for Nested Applications on page 42](#)
- [Configuring Global APPID Properties on page 43](#)
- [Configuring APPID Support for Heuristics on page 44](#)
- [Configuring APPID Support for Unidirectional Traffic on page 45](#)
- [Configuring Automatic Download of Application Package Updates on page 46](#)
- [Tracing APPID Operations on page 46](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

APPID Overview



NOTE: Starting with Junos OS Release 12.1, all interface-style services are supported for dynamic Point-to-Point Protocol over Ethernet (PPPoE) subscribers on all MX Series routers with modular Modular Port Concentrators (MPCs).

The APPID feature identifies applications as constituents of application groups in TCP/UDP/ICMP traffic. It is supported on MX Series routers equipped with Multiservices DPCs and on M120 or M320 routers equipped with Multiservices 400 PICs and Aggregated Multiservices (AMS) PICs. Aggregated Multiservices PICs (ams- interfaces) enable multiple ms- interfaces to be grouped together in a single bundle and cause the traffic destined for this AMS group to be distributed over the member services PICs of the group.

Junos OS Trio chipsets enable the calculation of a symmetric hash for the forward and reverse flows, and support a microcode map in the forwarding plane. This capability enables load-balancing of traffic across various services PICs in an AMS group. Starting with Junos OS Release 12.1, ams- interfaces enable an N:1 redundancy mechanism to cluster together N number of ms- interfaces in an AMS group that supports load sharing.



NOTE: For ams- interfaces and rms- interfaces, the statistics data in the bulk statistics file is collected using the reports received from the MS PICs. For the ams- interfaces, the retrieval and storage of statistics is not possible because of multiple PICs containing statistics data for the same subscriber. For interfaces in an AMS group, statistics data from different MS PICs in the AMS group are collected and aggregated on the Routing Engine where a timer control is activated and the data is saved in the bulkstats file based on this timer. This method of collection causes the statistics data in the bulkstats file to be displayed with a small delay period.

To configure APPID, include statements at the **[edit services application-identification]** hierarchy level to specify parameter values for defining applications, enable or disable application rules, and gather the applications and rules into groups.

The following are related operational commands:

- **show/clear application-identification application-system-cache**
- **show/clear application-identification counters**

For more information on the CLI configuration, see the *Application Identification*. For more information on the operational commands, see the [CLI Explorer](#).



NOTE: Because the extension-provider package framework lacks aggressive constraint checks, you should not set the `policy-db-size` statement at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level to a high value. For Junos Application Aware (previously known as Dynamic Application Awareness) configurations, the recommended values for the extension-provider package options at this hierarchy level are as follows:

- `control-cores = 1`
- `data-cores = 7`
- `object-cache-size = 1280` (for Multiservices 400 PIC and Multiservices DPC)
- `policy-db-size = 200`
- Include these package values: `jservices-idp`, `jservices-appid`, `jservices-llpdf`, `jservices-aacl`

For more information about this configuration, see the following topics in the *SDK Applications Configuration Guide and Command Reference*:

- *Configuring Control and Data Cores*
- *Configuring Memory Settings*
- *Configuring Packages on the PIC*



NOTE: In the export version of JUNOS, signature download is not expected to work for AppID and IDP features in Junos Application Aware. In order to make it work, you must additionally install the Crypto Software Suite.

Related Documentation

- [Defining an Application Identification on page 34](#)
- [Configuring APPID Rules on page 36](#)
- [Application Identification for Nested Applications on page 41](#)
- [Configuring Global APPID Properties on page 43](#)
- [Examples: Configuring Application Identification Properties on page 48](#)
- [\[edit services application-identification\] Hierarchy Level on page 72](#)

Best-Effort Application Identification of DPI-Serviced Flows

This topic describes the following information:

- [Features That Support Application-Level Filtering on page 32](#)
- [Best-Effort Application Determination on page 32](#)
- [APPID, AAACL, and L-PDF Processing in Preconvergence Scenarios on page 32](#)

Features That Support Application-Level Filtering

On MX Series routers equipped with Multiservices DPCs and M120 or M320 routers equipped with Multiservices 400 PICs, Intrusion Detection and Prevention (IDP) is accomplished by Deep Packet Inspection (DPI) of TCP, UDP, and ICMP flows. The application identification (APPID) feature defines applications as members of application groups in TCP/UDP/ICMP traffic. IDP depends on APPID for identification and detection of some Layer 7 applications.

The application-aware access list (AACL) service uses application names and groups as matching criteria for filtering traffic. The service defines the applications and application groups for which statistics are collected for a specific user or interface.

The local policy decision function (L-PDF) enables you to configure properties for statistics output. L-PDF supports a process that regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

Best-Effort Application Determination

Typically, APPID conclusively determines the Layer 7 application associated with a given DPI-serviced flow. In these cases, the application identification is final. Occasionally, APPID is only able to make an initial, inconclusive determination of the Layer 7 application associated with a given flow. This is referred to as a "best-effort" application identification. In such cases, the APPID process continues processing packets on that flow and might subsequently make a conclusive determination of the application associated with that flow. In some cases of best-effort application identification, the flow ends before a final application determination can be made.

APPID, AACL, and L-PDF Processing in Preconvergence Scenarios

The following sections describe APPID, AACL, and L-PDF processing in various stages of application identification for a DPI-serviced flow of TCP/UDP/ICMP traffic.

- [Prior to a Final or Best-Effort Application Identification on page 32](#)
- [Upon Best-Effort Application Identification on page 33](#)
- [While Application Identification Is on a Best-Effort Basis on page 33](#)
- [If a Flow Ends Before an Application Identification Is Made on page 33](#)
- [If a Flow Ends While Application Identification on a Best-Effort Basis on page 33](#)

Prior to a Final or Best-Effort Application Identification

During the time that APPID has not yet made either a final or best-effort determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has not yet made either a final or best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has not yet made either a final or best-effort determination of the associated application.

Upon Best-Effort Application Identification

When a best-effort application determination is made, AACL does not apply any AACL term actions configured for that flow. There are a number of reasons for this, one being that the action itself (such as "discard") could make a final application determination impossible. Instead, AACL or L-PDF tracks the flow and accepts all packets for that flow until a final determination is made, at which time the normal AACL or L-PDFL actions are fully applied to the flow.

While Application Identification Is on a Best-Effort Basis

During the time that APPID identification of the application associated with a given flow is on a best-effort basis, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has only made a best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has only made a best-effort determination of the associated application.

If a Flow Ends Before an Application Identification Is Made

If a flow ends before APPID has made either a final or a best-effort application identification, AACL or L-PDF uses the "unknown" application ID as a final determination and performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** AACL term action is configured for the "application-group-any" application, then the statistics for that flow will be collected and aggregated against the count bucket type, and reported as such.

If a Flow Ends While Application Identification on a Best-Effort Basis

If a flow ends while the application identification is on a best-effort basis, AACL or L-PDF uses that best-effort determination as a final determination. AACL or L-PDF performs

any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** AACL term action is configured for that Layer 7 application, then the statistics for the flow will be collected and aggregated against the AACL or L-PDF statistics. However, in the case of nested applications, AACL and L-PDF will not consider the best-effort determination as final and the nested application will be reported as an unknown application.

Related Documentation

- [Configuring AACL Rules on page 21](#)
- [Configuring Statistics Profiles on page 62](#)
- [aACL-fields on page 75](#)
- [aACL-statistics-profile on page 76](#)
- [rule on page 112](#)
- [services on page 120](#)
- [term on page 126](#)
- [then on page 127](#)

Defining an Application Identification

To configure a specific IP address or port-based application identification, include the **application** *application-name* statement at the **[edit services application-identification]** hierarchy level:

```
application application-name {  
  disable;  
  idle-timeout seconds;  
  index number;  
  session-timeout seconds;  
  type type;  
  type-of-service service-type;  
  port-mapping {  
    port-range {  
      tcp [ ports-and-port-ranges ];  
      udp [ ports-and-port-ranges ];  
    }  
    disable;  
  }  
}
```

You can include the following general properties in the configuration:

- **application**—Application name, a required statement; maximum 31 characters. Predefined applications have the prefix **junos-** to avoid conflict with user-defined ones.
- **idle-timeout**—Amount of time that a session remains idle before it is deleted.
- **index**—Application index number in the range from 1 through 65,534, with integers 1 through 1024 reserved for predefined applications.
- **session-timeout**—Lifetime of a session.

- **type**—Well known applications, such as HTTP or FTP.
- **type-of-service**—Type of service, defined by service objective. There is no default value; options are **maximize-reliability**, **maximize-throughput**, **minimize-delay**, and **minimize-monetary-cost**.
- **disable**—Disable this application definition in the APPID service.



NOTE: You can also specify session and idle timeout values globally for a Multiservices interface by including the following statements at the [edit interfaces *interface-name* services-options] hierarchy level:

- **inactivity-non-tcp-timeout**—Inactivity timeout period for non-TCP established sessions.
- **inactivity-tcp-timeout**—Inactivity timeout period for TCP established sessions.
- **session-timeout**—Lifetime of a session.
- **disable-global-timeout-override**—Disallow overriding a global inactivity or session timeout.

You can include the following port-mapping properties at the [edit services application-identification port-mapping] hierarchy level:

- **port-range**—TCP or UDP port number or numeric range, entered as [*minimum-value* – *maximum-value*]. For port-mapping configurations, this entry is required if the parent node exists.
- **disable**—Disable port-mapping properties for this application.



NOTE: For applications with signatures for both client-to-server and server-to-client directions, the APPID for Junos Application Aware (previously known as Dynamic Application Awareness) must accept the data packets in both directions on the same session to complete the identification process.

For a configuration example, see “[Examples: Configuring Application Identification Properties](#)” on page 48.

Related Documentation

- [APPID Overview on page 29](#)
- [Configuring APPID Rules on page 36](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 37](#)
- [Configuring Application Profiles on page 39](#)
- [Configuring Application Groups on page 40](#)
- [Tracing APPID Operations on page 46](#)

- [\[edit services application-identification\] Hierarchy Level on page 72](#)

Configuring APPID Rules

This configuration specifies the properties for identifying an application for which a source or destination IP address and port is used for a known application, without the requirement of an application signature. For example, the Session Initiation Protocol (SIP) server initiates a session from its identified port, 5060. You can therefore specify the SIP server IP address and port 5060 in the port mapping configuration for the SIP application. The advantage of using this method is to provide efficiency and accuracy of application identification for your network.

To configure application rule properties, include the **rule** statement at the [\[edit services application-identification\]](#) hierarchy level:

```
rule rule-name {  
  address address-name {  
    destination {  
      ip address</prefix-length>;  
      port-range {  
        tcp [ ports-and-port-ranges ];  
        udp [ ports-and-port-ranges ];  
      }  
    }  
    source {  
      ip address</prefix-length>;  
      port-range {  
        tcp [ ports-and-port-ranges ];  
        udp [ ports-and-port-ranges ];  
      }  
    }  
    order number;  
  }  
  application application-name;  
  disable;  
}
```

You can include the following application rule properties:

- **address**—Address properties for APPID rule processing. This statement is mandatory; you must specify either destination or source properties.
- **destination**—Destination address and port information. The **ip** statement defines the IP address and netmask (IPv4 only), and the **port-range** statement defines the TCP or UDP port number or numeric range, entered as **[minimum-value – maximum-value]**.
- **source**—Source address and port information. The **ip** statement defines the IP address and netmask (IPv4 only), and the **port-range** statement defines the TCP or UDP port number or numeric range, entered as **[minimum-value – maximum-value]**.
- **order**—Application matching priority. For address configurations, the order number resolves the conflict when multiple address entries are matched for a specific session;

the lower the number, the higher the priority. This statement is mandatory and must contain a unique value.

- **application**—Name of the application to be included in the rule.
- **disable**—Disable processing for this application rule.

The **rule-set** statement defines a collection of APPID rules that determine what actions the router software performs on packets in the data stream. You define each rule by specifying a rule name and configuring terms. Then, you specify the order of the rules by including the **rule-set** statement at the **[edit services application-identification]** hierarchy level with a **rule** statement for each rule:

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

**Related
Documentation**

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 37](#)
- [Configuring Application Profiles on page 39](#)
- [Configuring Application Groups on page 40](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Using Stateful Firewall Rules to Identify Data Sessions

The APPID configuration properties enable the Junos OS to detect applications based on signatures, ports, and addresses. For signature-based detection, most of the protocol control sessions are identified, but data sessions are not identified. For example, APPID identifies FTP connections to port 21 (FTP control sessions); however, FTP can open child/data sessions to transfer files and data. These sessions are not identified by signature-based APPID because they do not have well-defined signatures.

Application-level gateways (ALGs) configured using stateful firewall rules can assist APPID in identifying these data sessions. These sessions include file and video transfers that are heavy consumers of bandwidth, so a mechanism for policing and classifying this traffic effectively is a useful tool. In addition to FTP, this mechanism applies to TFTP and RTSP traffic.

To incorporate the stateful firewall rules into Junos Application Aware (previously known as Dynamic Application Awareness for Junos OS) sessions, include the following configurations:

1. Include the stateful firewall package at the **[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]** hierarchy level:

```
package jservices-sfw;
```
2. Define two stateful firewall rules as shown in the following example, one to identify the appropriate ALGs for FTP, TFTP, or RTSP traffic and the other to allow all traffic:



NOTE: Session Initiation Protocol (SIP) is already covered by APPID and the SIP ALG is not supported by stateful firewall, hence a SIP configuration is not needed.

```
[edit services]
stateful-firewall {
  rule rule1 {
    match-direction input-output;
    term term1 {
      from {
        applications [ junos-ftp junos-tftp junos-rtsp ];
      }
      then {
        accept;
      }
    }
  }
  rule rule2 {
    match-direction input-output;
    term term1 {
      then {
        accept;
      }
    }
  }
  rule-set rs1 {
    rule rule1;
    rule rule2;
  }
}
```



NOTE: The existing AACL and L-PDF operational mode commands should report the new applications when they are identified.

3. Attach the stateful firewall rule set to a service set, as shown in the following example:

```
service-set test-chaining {
  application-identification-profile add-based;
  stateful-firewall-rule-sets rs1;
  idp-profile idp1;
  aacl-rules rule1;
  interface-service {
    service-interface ms-2/0/0.0;
  }
}
```

4. Include *no-drop* settings for stateful firewall and TCP, as needed.

Stateful firewall processing drops packets in a number of scenarios:

- TCP sessions do not start with a SYN flag. (This prevents sessions from resuming; otherwise, when the PIC starts for the first time, all existing TCP sessions in flight will be dropped).
- If the TCP tracker detects SYN but no SYN/ACK or only an ACK, then the ACK is dropped. There are a number of similar checks to verify the TCP connection, window checks, and so forth.
- TCP checks for stateful firewall are aggressive when ALGs are run. It is not possible to ignore TCP errors when an ALG is run on a session.
- If an ALG detects malformed packets (for example, if the FTP PORT command is not RFC-compliant), it drops packets. If an ALG is not able to allocate resources, it drops packets.

You can include the settings shown in the following example to assist in controlling these packet drops:

```
[edit interfaces]
ms-1/2/0 {
  services-options {
    ignore-errors {
      tcp;
      alg;
    }
  }
}
```

The **tcp** statement mediates the first two issues listed, with reference to TCP SYN detection. The **alg** statement handles the fourth issue. ALGs require strict TCP processing, which cannot be relaxed.

**Related
Documentation**

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Application Identification for Nested Applications on page 41](#)
- [Configuring Global APPID Properties on page 43](#)
- [Tracing APPID Operations on page 46](#)

Configuring Application Profiles

You can define an application profile for use in a service set. The profile consists of one or more rule sets, but only one profile can be included per service set.

To specify the application profile constituents, include the **profile** statement at the **[edit services application-identification]** hierarchy level:

```
profile profile-name {
  [ rule-set rule-set-name ];
}
```

You assign a profile name and include one or more predefined rule sets. For more information on rule sets, see [“Configuring APPID Rules” on page 36](#). You can then include the profile in a service-set definition:

```
[edit services]
service-set service-set-name {
  profile profile-name;
}
```

The definitions specific to Junos Application Aware (previously known as Dynamic Application Awareness) include the APPID and IDP profiles and the ACL rule set. For more information on service sets, see *Service Set Properties*.

- Related Documentation**
- [APPID Overview on page 29](#)
 - [Defining an Application Identification on page 34](#)
 - [Configuring Application Groups on page 40](#)
 - [Configuring Global APPID Properties on page 43](#)

Configuring Application Groups

You can define an application group to process a number of applications or subgroups at the same time. To configure application group properties, include the **application-group** statement at the **[edit services application-identification]** hierarchy level:

```
application-group group-name {
  application-groups {
    application-group-name;
  }
  applications {
    application-name;
  }
  index number;
  disable;
}
```

You can include the following application group properties:

- **applications**—List of applications to include in this application group. The **name** statement is mandatory and must include at least one entry.
- **application-groups**—List of application groups to include in a larger application group. The **name** statement is mandatory and must include at least one entry.
- **index**—Application group index number in the range from 1 through 65,534. This mandatory value must be unique.
- **disable**—Disable processing for this application group.

- Related Documentation**
- [Defining an Application Identification on page 34](#)
 - [Configuring APPID Rules on page 36](#)
 - [Configuring Application Profiles on page 39](#)

- [Configuring Global APPID Properties on page 43](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Application Identification for Nested Applications

The application identification feature is used by intrusion detection and prevention (IDP) to allow or deny traffic based on applications running on standard or nonstandard ports. *Nested applications* are protocols running over the parent application. For example, both Facebook and Yahoo Messenger can run over HTTP, but there is a need to identify them as two different applications. To do this, the application layer is split into two layers: Layer 7 applications and Layer 7 protocols.

The predefined application signatures included with Junos OS have been created to detect the Layer 7 nested applications. Predefined application signatures can be used in attack objects.

To configure nested application properties, include the **nested-application** statement at the **[edit services application-identification]** hierarchy level:

```
nested-application name {
  index number;
  protocol protocol;
  signature name {
    chain-order;
    maximum-transactions number;
    member name {
      context (http-header-content-type | http-header-host | http-url-parsed |
        http-url-parsed-param-parsed);
      direction (any | client-to-server | server-to-client);
      pattern dfa-pattern;
    }
    order number;
  }
  type type;
}
```

You can include the following application rule properties:

- **chain-order**—Signatures can contain multiple members. If the chain order feature is on, those members are read in order. The default for this option is no chain order. If a signature contains only one member, this option is ignored.
- **context**—Define a service specific context. The options are **http-header-content-type**, **http-header-host**, **http-url-parsed**, **http-url-parsed-param-parsed**. This statement is mandatory.
- **direction**—The connection direction of the packets to apply pattern matching. The options are **client-to-server**, **server-to-client**, or **any**. This statement is mandatory.
- **index**—A number that is a one-to-one mapping to the application name that is used to ensure that each signature definition is unique. The index range for predefined

applications is 1 through 32767. The index range for custom applications and custom nested applications is 32768 through 65534.

- **maximum transactions**—The maximum number of transactions that should occur before a match is made. This statement is mandatory.
- **member**—Define a member name for a custom nested application signature definition. Custom definitions can contain multiple members that define attributes for an application.
- **order**—Define application matching priority. For address configurations, the order number resolves the conflict when multiple address entries are matched for a specific session. The lower number has higher priority. This statement is mandatory.
- **pattern**—Define an attack pattern to be detected. This statement is mandatory.
- **protocol**—The protocol that will be monitored to identify nested applications. The value **http** is supported. This statement is mandatory.
- **signature**—Name of the custom nested application signature definition. Must be a unique name with a maximum length of 32 characters. This statement is mandatory.
- **type**—Well-known application name for this application definition, such as Facebook or Kazza. This application name must be unique with a maximum length of 32 characters. This statement is mandatory.

**Related
Documentation**

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Disabling Application Identification for Nested Applications on page 42](#)
- [Configuring Global APPID Properties on page 43](#)
- [Tracing APPID Operations on page 46](#)

Disabling Application Identification for Nested Applications

Sometimes there is a need to identify multiple different applications running on the same Layer 7 protocols. For example, both Facebook and Yahoo Messenger can run over HTTP, but there is a need to identify them as two different applications. To do this, the application layer is split into two layers: Layer 7 applications and Layer 7 protocols. Application identification for nested applications is turned on by default. You can manually turn it off by using the CLI.

To disable nested application identification:

- Set the **no-nested-application** statement.

```
[edit services application-identification nested-application-settings]  
user@host# no-nested-application
```

To verify the configuration, issue the **show services application-identification nested-application-settings** command.

To reenable nested application identification:

- Delete the **no-nested-application** statement.

```
[edit services application-identification nested-application-settings]
user@host# delete services application-identification nested-application-settings
no-nested-application
```

If you are finished configuring the device, commit the configuration.

**Related
Documentation**

- [APPID Overview on page 29](#)
- [Application Identification for Nested Applications on page 41](#)

Configuring Global APPID Properties

You can define additional properties that apply on a global basis to APPID processing and are not part of a specific application, group, rule, or profile definition. To configure these global APPID properties, include the following statements at the **[edit services application-identification]** hierarchy level:

```
application-identification {
  application-system-cache-timeout seconds;
  max-checked-bytes bytes;
  min-checked-bytes bytes;
  nested-application name
  nested-application-settings
  no-application-identification
  no-application-system-cache;
  no-clear-application-system-cache;
  no-protocol-method;
  no-signature-based;
  signature-method-all-ports;
}
```

The global application properties have the following effect:

- **application-system-cache-timeout**—Lifetime for system cache entries, in seconds.
- **max-checked-bytes**—The maximum number of bytes to be inspected in APPID processing, in the range from 0 through 100,000 bytes.
- **min-checked-bytes**—The minimum number of bytes to be inspected in APPID processing, in the range from 0 through 2000 bytes.
- **nested-application**—Configure a custom nested application definition for the desired application name that will be used by the system to identify the nested application as it passes through the device. For more information see [nested-application](#).
- **nested-application-settings**—Configure nested application options for application identification services. For more information see [nested-application-settings](#).
- **no-application-identification**—Disable all application identification methods.
- **no-application-system-cache**—Disable storing application identification results in the application system cache.

- **no-clear-application-system-cache**—Disable clearing the application system cache.
- **no-protocol-method**—Disable the protocol-based application identification method, which is enabled by default.
- **no-signature-based**—Disable the signature-based application identification method.
- **signature-method-all-ports**—Run signature matching on all traffic.

Related Documentation

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Application Identification for Nested Applications on page 41](#)
- [Disabling Application Identification for Nested Applications on page 42](#)
- [Tracing APPID Operations on page 46](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Configuring APPID Support for Heuristics

Heuristics methodology provides a mechanism for identifying encrypted data packets in point-to-point applications. These packets are not normally detected by the existing application signatures.

To enable APPID to employ heuristics in traffic identification:

1. Include the **enable-heuristics** statement:

```
[edit services application-identification]  
user@host# enable-heuristics
```

The **show services application-identification counter** operational command includes additional output fields that report the number of encrypted sessions.



NOTE: When you enable heuristics, performance and scaling values might be negatively affected. This mechanism assists the APPID module in identifying encrypted traffic, but only if the identifications are supported by the current signature package.

Related Documentation

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Application Identification for Nested Applications on page 41](#)
- [Configuring Global APPID Properties on page 43](#)
- [Configuring APPID Support for Unidirectional Traffic on page 45](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Configuring APPID Support for Unidirectional Traffic

With asymmetrical routing, a networking device sees only one side of the network sessions, either from client to server or from server to client. Additional functionality is required to support application identification with unidirectional traffic. This addition enables a session for a specified service set to support an asymmetrical routing environment, and allows complete application matches using existing application signatures for traffic in the client-to-server direction only.

To enable APPID to support application matching on unidirectional traffic:

1. Include the **support-uni-directional-traffic** statement:

```
[edit services service-set service-set-name service-set-options]
user@host# support-uni-directional-traffic
```

This enables the session belonging to the specified service set to support the asymmetrical routing environment. The APPID module then reports complete matches for the unidirectional traffic.

2. Include the **enable-asymmetric-traffic-processing** statement:

```
[edit services service-set service-set-name service-set-options]
user@host# enable-asymmetric-traffic-processing
```

This enables the framework and plug-in to handle unidirectional traffic at a service-set level.

When you enable these settings, APPID treats unidirectional TCP traffic like a UDP connection. UDP traffic itself does not receive any special treatment because the service PIC cannot determine whether UDP traffic is unidirectional or bidirectional. The settings do not affect processing of sessions created with bidirectional traffic.

If the traffic includes both unidirectional and bidirectional sessions, the APPID module uses heuristics to decide whether to change the reporting logic.



NOTE: This feature does not change the processing for any services except APPID. However, other services, including stateful firewall, AACL, and IDP, can process unidirectional traffic in a limited manner.

Related Documentation

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Application Identification for Nested Applications on page 41](#)
- [Configuring Global APPID Properties on page 43](#)
- [Configuring APPID Support for Heuristics on page 44](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Configuring Automatic Download of Application Package Updates

You can set up automatic downloading of application package updates. To configure downloads, include the **download** statement at the **[edit services application-identification]** hierarchy level:

```
download {  
  automatic {  
    interval hour;  
    start-time time;  
  }  
  url url;  
}
```

You can include the following download statements:

- **download**—Define download properties.
- **automatic**—Set **start-time** value and **interval** in hours for automatic downloads. The default **start-time** is **0:00** and the range is from 0:00 through 24:00. The default **interval** is **24** and the range is from 1 through 168.
- **url**—Specify the download URL.

Related Documentation

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Tracing APPID Operations

Tracing operations track all adaptive services operations and record them in a log file. The logged error descriptions provide detailed information to help you solve problems faster.

By default, no events are traced. If you include the **traceoptions** statement at the **[edit services application-identification]** hierarchy level, the default tracing behavior is as follows:

- Important events are logged in a file called **serviced** located in the **/var/log** directory.
- When the file **serviced** reaches 128 kilobytes (KB), it is renamed **serviced.0**, then **serviced.1**, and so on, until there are three trace files. Then the oldest trace file (**serviced.2**) is overwritten. (For more information about how log files are created, see the [System Log Explorer](#).)
- Only the user who configures the tracing operation can access the log files.
- To display the end of the log, issue the **show log serviced | last** operational mode command:

```
[edit]  
user@host# run show log serviced | last
```

You cannot change the directory (**/var/log**) in which trace files are located. However, you can customize the other trace file settings by including the following statements:

```
file filename <files number> <match regex> <size size> <(world-readable |  
no-world-readable);  
flag {  
    all;  
}
```

You configure these statements at the **[edit services application-identification traceoptions]** hierarchy level.

These statements are described in the following sections:

- [Configuring the APPID Log Filename on page 47](#)
- [Configuring the Number and Size of APPID Log Files on page 47](#)
- [Configuring Access to the Log File on page 47](#)
- [Configuring a Regular Expression for Lines to Be Logged on page 48](#)
- [Configuring the Tracing Flags on page 48](#)

Configuring the APPID Log Filename

By default, the name of the file that records trace output is **serviced**. You can specify a different name by including the **file** statement at the **[edit services application-identification traceoptions]** hierarchy level:

```
file filename;
```

Configuring the Number and Size of APPID Log Files

By default, when the trace file reaches 128 kilobytes (KB) in size, it is renamed **filename.0**, then **filename.1**, and so on, until there are three trace files. Then the oldest trace file (**filename.2**) is overwritten.

You can configure the limits on the number and size of trace files by including the following statements at the **[edit services application-identification traceoptions]** hierarchy level:

```
file files number size size;
```

For example, set the maximum file size to 2 MB, and the maximum number of files to 20. When the file that receives the output of the tracing operation (**filename**) reaches 2 MB, **filename** is renamed **filename.0**, and a new file called **filename** is created. When the new **filename** reaches 2 MB, **filename.0** is renamed **filename.1** and **filename** is renamed **filename.0**. This process repeats until there are 20 trace files. Then the oldest file (**filename.19**) is overwritten by the newest file (**filename.0**).

The number of files can be from 2 through 1000 files. The file size of each file can be from 10 KB through 1 gigabyte (GB).

Configuring Access to the Log File

By default, only the user who configures the tracing operation can access log files.

To specify that any user can read all log files, include the **file world-readable** statement at the **[edit services application-identification traceoptions]** hierarchy level:

```
file world-readable;
```

To explicitly set the default behavior, include the **file no-world-readable** statement at the **[edit services application-identification traceoptions]** hierarchy level:

```
file no-world-readable;
```

Configuring a Regular Expression for Lines to Be Logged

By default, the trace operation output includes all lines relevant to the logged events.

You can refine the output by including the **match** statement at the **[edit services application-identification traceoptions file filename]** hierarchy level and specifying a regular expression (regex) to be matched:

```
file filename match regex;
```

Configuring the Tracing Flags

By default, if the **traceoptions** configuration is present, only important events are logged. You can configure the trace operations to be logged by including the following statements at the **[edit services application-identification traceoptions]** hierarchy level:

```
flag {  
  all;  
}
```

Currently, the only supported flag is **all**, which instructs the router to trace all operations.

Related Documentation

- [APPID Overview on page 29](#)
- [Defining an Application Identification on page 34](#)
- [Examples: Configuring Application Identification Properties on page 48](#)

Examples: Configuring Application Identification Properties

The following examples show an address-based application identification configuration:

```
[edit services application-identification]  
rule rule1 {  
  application-name test2;  
  address 1 {  
    source {  
      ip 10.110.1.1/16;  
      port-range {  
        tcp 1110-1150;  
      }  
    }  
  }  
  destination {  
    ip 10.11.1.1/16;  
    port-range {  
      tcp 111-1100;  
    }  
  }  
}
```



```
    }  
  }  
  order 1;  
}  
}  
}  
[edit services application-identification]  
rule-set rs1 {  
  rule rule1;  
}  
profile pf1 {  
  rule-set rs1;  
}  
[edit services]  
service-set sset1 {  
  application-identification-profile pf1;  
}
```

The following examples show application group configuration:

```
[edit services application-identification]  
application-group junos:peer-to-peer {  
  index 5;  
  application-groups {  
    junos:chat;  
    junos:file-sharing;  
    junos:voip;  
  }  
}  
[edit services application-identification]  
application-group junos:voip {  
  index 14;  
  applications {  
    junos:h225ras;  
    junos:h225sgn;  
    junos:mgcp;  
    junos:sip;  
  }  
}
```

The following examples show application identification for nested application configuration:

```
nested-application nested1 {  
  type nested1;  
  index 65345;  
  protocol HTTP;  
  signature nestedcust001 {  
    member m01 {  
      context http-url-parsed;  
      pattern .*nested.*;  
      direction any;  
    }  
    maximum-transactions 2;  
    order 3825;  
  }
```


CHAPTER 3

Detecting Suspicious and Anomalous Network Traffic Using IDP

- [IDP Overview on page 51](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 53](#)
- [Configuring Multiple IDP Detectors on page 56](#)

IDP Overview



NOTE: Starting with Junos OS Release 12.1, all interface-style services are supported for dynamic Point-to-Point Protocol over Ethernet (PPPoE) subscribers on all MX Series routers with modular Modular Port Concentrators (MPCs).

The Junos Application Aware (previously known as Dynamic Application Awareness for the Junos OS) set of services adds support for the intrusion detection and prevention (IDP) functionality using Deep Packet Inspection (DPI) technology to Juniper Networks MX Series 3D Universal Edge Routers equipped with Multiservices Dense Port Concentrators (MS-DPCs) and M120 or M320 Multiservices Edge Routers equipped with Multiservices 400 PICs.

The IDP functionality is already supported on Juniper Networks SRX Series Services Gateways running Junos OS and is described in the *Junos OS Intrusion Detection and Prevention (IDP) Library for Security Devices*. Starting with Junos OS Release 11.3, support for the IDP functionality is extended to T320, T640, and T1600 routers. In addition, multiple IDP detectors are now supported on the M120, M320, and MX Series routers with Enhanced III Flexible PIC Concentrators (FPCs).



NOTE: In the export version of Junos OS, signature download is not expected to work for AppID and IDP features in Junos Application Aware. In order to make it work, you must additionally install the Crypto Software Suite.

The same CLI statements and commands are used on all platforms with the following caveats:

- **Service sets**—IDP is incorporated as a component of service sets only on the specified Juniper Networks T Series, M Series and MX Series routers. IDP depends on application identification services (APPID) for definition and detection of some Layer 7 applications. Before configuring an IDP policy, you must download the APPID application package. Only one service set can be applied to a single interface when the APPID functionality is used.
- **Multiple IDP detectors**—Except for the maximum number of decoder binary instances (4) that are loaded into the process space, multiple IDP detectors on the M120, M320, and MX Series routers function in a similar way to the existing IDP detector support on SRX Series devices. To view the current policy and the corresponding detector version, use the **show security idp status detail** command.

To configure IDP properties, include statements at the **[edit security idp]** hierarchy level. In general, you configure IDP processes by including the **idp-policy** statement at the **[edit system processes]** hierarchy level. For use in T Series, M Series and MX Series applications, you then reference this configuration by including the **idp-profile** statement at the **[edit services service-set]** hierarchy level. To configure SNMP IDP objects, include the **idp** statement at the **[edit snmp health-monitor]** hierarchy level. The operational commands for monitoring and regulating IDP activity are the **clear security idp**, **request security idp**, and **show security idp** commands.

To configure the source IP address for downloading security packages, use the command **set security idp security-package source-address ip-address** because it is not possible to download security packages if the router uses private addressing on its outgoing interface. The source address should be a valid IP address on the node.



NOTE: On T Series, M Series and MX Series routers, the IDP **ip-action** statement is supported on TCP, UDP, and ICMP flows. When the **ip-action target** is **service**, the **ip-action** flow is applied if the traffic matches the values specified for the source port, destination port, source address, and destination address. However, for ICMP flows, the destination port is 0, so that any ICMP flow matching the source port, source address, and destination address would be blocked. For more information about the **ip-action** statement, see the *Junos OS CLI Reference*.

When the Multiservices PIC configured for a service set is either administratively taken offline or undergoes a failure, all the traffic entering the configured interface with an IDP service set would be dropped without notification. To avoid this traffic loss, include the **bypass-traffic-on-pic-failure** statement at the **[edit services service-set service-set-name service-set-options]** hierarchy level. When this statement is configured, the affected packets are forwarded in the event of a Multiservices PIC failure or offlining, as though interface-style services were not configured.



NOTE: Data channel applications for protocols such as FTP, TFTP, RTSP, and SIP are not in the same application group as their control channel applications. For example, control channel application `junos:ftp` is in the group `junos:file-server` but the corresponding data application `junos:system:ftp-data` is not in any group.



NOTE: Because the extension-provider package framework lacks aggressive constraint checks, you should not set the `policy-db-size` statement at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level to a high value. For Junos Application Aware configurations, the recommended values for the extension-provider package options at this hierarchy level are as follows:

- `control-cores = 1`
- `data-cores = 7`
- `object-cache-size = 1280` (for Multiservices 400 PIC and Multiservices DPC)
- `policy-db-size = 200`
- Include these package values: `jservices-idp`, `jservices-appid`, `jservices-llpdf`, `jservices-aacl`

For more information about this configuration, see the following topics in the *SDK Applications Configuration Guide and Command Reference*:

- *Configuring Control and Data Cores*
- *Configuring Memory Settings*
- *Configuring Packages on the PIC*

Related Documentation

- [Configuring Multiple IDP Detectors on page 56](#)

Best-Effort Application Identification of DPI-Serviced Flows

This topic describes the following information:

- [Features That Support Application-Level Filtering on page 53](#)
- [Best-Effort Application Determination on page 54](#)
- [APPID, AAACL, and L-PDF Processing in Preconvergence Scenarios on page 54](#)

Features That Support Application-Level Filtering

On MX Series routers equipped with Multiservices DPCs and M120 or M320 routers equipped with Multiservices 400 PICs, Intrusion Detection and Prevention (IDP) is accomplished by Deep Packet Inspection (DPI) of TCP, UDP, and ICMP flows. The

application identification (APPID) feature defines applications as members of application groups in TCP/UDP/ICMP traffic. IDP depends on APPID for identification and detection of some Layer 7 applications.

The application-aware access list (AACL) service uses application names and groups as matching criteria for filtering traffic. The service defines the applications and application groups for which statistics are collected for a specific user or interface.

The local policy decision function (L-PDF) enables you to configure properties for statistics output. L-PDF supports a process that regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

Best-Effort Application Determination

Typically, APPID conclusively determines the Layer 7 application associated with a given DPI-serviced flow. In these cases, the application identification is final. Occasionally, APPID is only able to make an initial, inconclusive determination of the Layer 7 application associated with a given flow. This is referred to as a "best-effort" application identification. In such cases, the APPID process continues processing packets on that flow and might subsequently make a conclusive determination of the application associated with that flow. In some cases of best-effort application identification, the flow ends before a final application determination can be made.

APPID, AACL, and L-PDF Processing in Preconvergence Scenarios

The following sections describe APPID, AACL, and L-PDF processing in various stages of application identification for a DPI-serviced flow of TCP/UDP/ICMP traffic.

- [Prior to a Final or Best-Effort Application Identification on page 54](#)
- [Upon Best-Effort Application Identification on page 55](#)
- [While Application Identification Is on a Best-Effort Basis on page 55](#)
- [If a Flow Ends Before an Application Identification Is Made on page 55](#)
- [If a Flow Ends While Application Identification on a Best-Effort Basis on page 55](#)

Prior to a Final or Best-Effort Application Identification

During the time that APPID has not yet made either a final or best-effort determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has not yet made either a final or best-effort determination of the associated application:

- **show services local-policy-decision-function flows (interface *interface-name* | subscriber *subscriber-name*)**
- **show services application-aware-access-list flows (interface *interface-name* | subscriber *subscriber-name*)**

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has not yet made either a final or best-effort determination of the associated application.

Upon Best-Effort Application Identification

When a best-effort application determination is made, ACL does not apply any ACL term actions configured for that flow. There are a number of reasons for this, one being that the action itself (such as "discard") could make a final application determination impossible. Instead, ACL or L-PDF tracks the flow and accepts all packets for that flow until a final determination is made, at which time the normal ACL or L-PDF actions are fully applied to the flow.

While Application Identification Is on a Best-Effort Basis

During the time that APPID identification of the application associated with a given flow is on a best-effort basis, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has only made a best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has only made a best-effort determination of the associated application.

If a Flow Ends Before an Application Identification Is Made

If a flow ends before APPID has made either a final or a best-effort application identification, ACL or L-PDF uses the "unknown" application ID as a final determination and performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** ACL term action is configured for the "application-group-any" application, then the statistics for that flow will be collected and aggregated against the count bucket type, and reported as such.

If a Flow Ends While Application Identification on a Best-Effort Basis

If a flow ends while the application identification is on a best-effort basis, ACL or L-PDF uses that best-effort determination as a final determination. ACL or L-PDF performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** ACL term action is configured for that Layer 7 application, then the statistics for the flow will be collected and aggregated against the ACL or L-PDF statistics. However, in the case of nested applications, ACL and L-PDF will not consider the best-effort determination as final and the nested application will be reported as an unknown application.

- Related Documentation**
- [Configuring ACL Rules on page 21](#)
 - [Configuring Statistics Profiles on page 62](#)
 - [aACL-fields on page 75](#)
 - [aACL-statistics-profile on page 76](#)
 - [rule on page 112](#)
 - [services on page 120](#)
 - [term on page 126](#)
 - [then on page 127](#)

Configuring Multiple IDP Detectors

To configure multiple IDP detectors:

1. In configuration mode, go to the **[edit security]** hierarchy level:

```
user@host# edit security
```

2. Go to the **[edit security idp sensor-configuration flow]** hierarchy level:

```
[edit security]
user@host# edit idp sensor-configuration flow
```

3. Configure the **no-reset-on-policy** statement:

```
[edit security idp sensor-configuration flow]
user@host# set no-reset-on-policy
```

4. Verify the configuration:

```
[edit security]
user@host# show security
idp {
    sensor-configuration {
        flow {
            no-reset-on-policy;
        }
    }
}
```

- Related Documentation**
- [IDP Overview on page 51](#)

CHAPTER 4

Collecting Statistics and Tracking Data Using L-PDF

- [L-PDF Overview on page 57](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 59](#)
- [Configuring Statistics Profiles on page 62](#)
- [Applying L-PDF Profiles to Service Sets on page 65](#)
- [Tracing L-PDF Operations on page 67](#)

L-PDF Overview



NOTE: Starting with Junos OS Release 12.1, all interface-style services are supported for dynamic Point-to-Point Protocol over Ethernet (PPPoE) subscribers on all MX Series routers with modular Modular Port Concentrators (MPCs).

Starting with Junos OS Release 12.1, the local policy decision function (L-PDF) plug-in can offload flows to the Packet Forwarding Engine. Offloading is supported only on MX Series routers with Modular Port Concentrators (MPCs) and accomplished using the Juniper Forwarding Mechanism (JFM). JFM allows services flows to be offloaded to the Packet Forwarding Engine. However, 5-tuple flows cannot be offloaded. Apart from the local L-PDF plug-in, offloading is supported on the packet-triggered subscribers and policy control (PTSP) plug-in. The `show services application-aware-access-list flows subscriber subscriber-name` command displays offload status.

Local policy decision functionality for application-related services adds support for a new process that regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces. This functionality is collectively named the local policy decision function (L-PDF). L-PDF is supported on:

- MX Series routers equipped with Multiservices DPCs.
- M120 or M320 routers equipped with Multiservices 400 PICs.

- Aggregated Multiservices (AMS) PICs.

Multiple `ms-` interfaces can be bundled together in an AMS PIC interface, which causes the traffic destined for this AMS group to be distributed over the member services PICs of the group. Junos OS Trio chipsets enable the calculation of a symmetric hash for the forward and reverse flows, and support a microcode map in the forwarding plane. This capability enables load-balancing of traffic across various services PICs in an AMS group. Starting with Junos OS Release 12.1, `ams-` interfaces enable an N:1 redundancy mechanism to cluster together N number of **`ms-` interfaces** in an AMS group that supports load sharing.

Starting with Junos OS Release 11.3, local L-PDF that resides on the services PIC is supported on T320, T640, and T1600 routers. The application identification (APPID) service defines the applications and how they are grouped. The application-aware access list (AACL) service defines the applications and application groups for which statistics are collected for a specific user or interface. The L-PDF configuration defines the way in which the statistics are output.

To configure properties for statistics output, include the **`policy-decision-statistics-profile`** statement at the **`[edit accounting-options]`** hierarchy level. A new **`traceoptions`** configuration is available at the **`[edit system services local-policy-decision-function]`** hierarchy level. To configure a dynamic profile to attach a specified service set to an interface, include the **`service`** statement at the **`[edit dynamic-profiles profile-name interfaces interface-name unit logical-unit-number family inet]`** hierarchy level. To attach a service set to a static interface, include the **`service-set service-set-name`** statement at the **`[edit interfaces interface-name unit logical-unit-number family inet service (input | output)]`** hierarchy level. For more information on service sets, see *Service Set Properties*.

The following related operational commands are supported:

- **`show services local-policy-decision-function flows`**
- **`show/clear services local-policy-decision-function statistics`**
- **`show/clear services application-aware-access-list statistics`**

For more information on the CLI configuration, see the *Local Policy Decision Function*. For more information on the operational commands, see the [CLI Explorer](#).



NOTE: Because the Junos OS extension-provider package (variously known as JSF, MP-SDK, and eJunos in releases earlier than 12.3) lacks aggressive constraint checks, you should not set the `policy-db-size` statement at the [edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider] hierarchy level to a high value. For Junos Application Aware (previously known as Dynamic Application Awareness) configurations, the recommended values for the extension-provider package options at this hierarchy level are as follows:

- `control-cores = 1`
- `data-cores = 7`
- `object-cache-size = 1280` (for Multiservices 400 PIC and Multiservices DPC)
- `policy-db-size = 200`
- Include these package values: `jservices-idp`, `jservices-appid`, `jservices-llpdf`, `jservices-aacl`

For more information about this configuration, see the following topics in the *SDK Applications Configuration Guide and Command Reference*:

- *Configuring Control and Data Cores*
- *Configuring Memory Settings*
- *Configuring Packages on the PIC*

Related Documentation

- [Best-Effort Application Identification of DPI-Serviced Flows on page 18](#)
- [Configuring Statistics Profiles on page 62](#)
- [Applying L-PDF Profiles to Service Sets on page 65](#)
- [Tracing L-PDF Operations on page 67](#)

Best-Effort Application Identification of DPI-Serviced Flows

This topic describes the following information:

- [Features That Support Application-Level Filtering on page 59](#)
- [Best-Effort Application Determination on page 60](#)
- [APPID, AAACL, and L-PDF Processing in Preconvergence Scenarios on page 60](#)

Features That Support Application-Level Filtering

On MX Series routers equipped with Multiservices DPCs and M120 or M320 routers equipped with Multiservices 400 PICs, Intrusion Detection and Prevention (IDP) is accomplished by Deep Packet Inspection (DPI) of TCP, UDP, and ICMP flows. The application identification (APPID) feature defines applications as members of application

groups in TCP/UDP/ICMP traffic. IDP depends on APPID for identification and detection of some Layer 7 applications.

The application-aware access list (AACL) service uses application names and groups as matching criteria for filtering traffic. The service defines the applications and application groups for which statistics are collected for a specific user or interface.

The local policy decision function (L-PDF) enables you to configure properties for statistics output. L-PDF supports a process that regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

Best-Effort Application Determination

Typically, APPID conclusively determines the Layer 7 application associated with a given DPI-serviced flow. In these cases, the application identification is final. Occasionally, APPID is only able to make an initial, inconclusive determination of the Layer 7 application associated with a given flow. This is referred to as a "best-effort" application identification. In such cases, the APPID process continues processing packets on that flow and might subsequently make a conclusive determination of the application associated with that flow. In some cases of best-effort application identification, the flow ends before a final application determination can be made.

APPID, AACL, and L-PDF Processing in Preconvergence Scenarios

The following sections describe APPID, AACL, and L-PDF processing in various stages of application identification for a DPI-serviced flow of TCP/UDP/ICMP traffic.

- [Prior to a Final or Best-Effort Application Identification on page 60](#)
- [Upon Best-Effort Application Identification on page 61](#)
- [While Application Identification Is on a Best-Effort Basis on page 61](#)
- [If a Flow Ends Before an Application Identification Is Made on page 61](#)
- [If a Flow Ends While Application Identification on a Best-Effort Basis on page 61](#)

Prior to a Final or Best-Effort Application Identification

During the time that APPID has not yet made either a final or best-effort determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has not yet made either a final or best-effort determination of the associated application:

- **show services local-policy-decision-function flows (interface *interface-name* | subscriber *subscriber-name*)**
- **show services application-aware-access-list flows (interface *interface-name* | subscriber *subscriber-name*)**

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has not yet made either a final or best-effort determination of the associated application.

Upon Best-Effort Application Identification

When a best-effort application determination is made, ACL does not apply any ACL term actions configured for that flow. There are a number of reasons for this, one being that the action itself (such as "discard") could make a final application determination impossible. Instead, ACL or L-PDF tracks the flow and accepts all packets for that flow until a final determination is made, at which time the normal ACL or L-PDF actions are fully applied to the flow.

While Application Identification Is on a Best-Effort Basis

During the time that APPID identification of the application associated with a given flow is on a best-effort basis, the flow does not contribute to any per-subscriber or per-application statistics collection.

The output of the following operational mode commands includes flows for which APPID has only made a best-effort determination of the associated application:

- **show services local-policy-decision-function flows** (interface *interface-name* | subscriber *subscriber-name*)
- **show services application-aware-access-list flows** (interface *interface-name* | subscriber *subscriber-name*)

In the command output, the **Action** field displays "accept" and the **Application** or **Application group** field displays "unknown" for a flow for which APPID has only made a best-effort determination of the associated application.

If a Flow Ends Before an Application Identification Is Made

If a flow ends before APPID has made either a final or a best-effort application identification, ACL or L-PDF uses the "unknown" application ID as a final determination and performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** ACL term action is configured for the "application-group-any" application, then the statistics for that flow will be collected and aggregated against the count bucket type, and reported as such.

If a Flow Ends While Application Identification on a Best-Effort Basis

If a flow ends while the application identification is on a best-effort basis, ACL or L-PDF uses that best-effort determination as a final determination. ACL or L-PDF performs any necessary collection, aggregation, and reporting of statistics based on that Layer 7 application. In particular, if the **count** ACL term action is configured for that Layer 7 application, then the statistics for the flow will be collected and aggregated against the ACL or L-PDF statistics. However, in the case of nested applications, ACL and L-PDF will not consider the best-effort determination as final and the nested application will be reported as an unknown application.

Related Documentation

- [Configuring ACL Rules on page 21](#)
- [Configuring Statistics Profiles on page 62](#)
- [acl-fields on page 75](#)

- [aocl-statistics-profile on page 76](#)
- [rule on page 112](#)
- [services on page 120](#)
- [term on page 126](#)
- [then on page 127](#)

Configuring Statistics Profiles

The local policy decision function (L-PDF) enables you to configure properties for statistics output. To do this, you create a statistics profile, which configures the files to which statistics records are exported and the format that is exported. There are two configurations you can use to specify the profile, as described in the following subsections:

- [Configuring an L-PDF Statistics Profile on page 63](#)
- [Configuring an ACL Statistics Profile on page 64](#)



NOTE: You must use the same configuration stanza for specifying the profile and the file selection. If configurations are committed in both hierarchies, the one at the `[edit system services local-policy-decision-function]` hierarchy level takes precedence.



NOTE:

- When a session closes before APPID has identified nested applications, the session is treated as a best-effort session and L-PDF does not get the nested application information. In such cases, nested applications will be reported as unknown applications.
- During the time that the application identification (APPID) feature has not yet made a final determination of the application associated with a given flow, the flow does not contribute to any per-subscriber or per-application statistics collection. For more information, see [“Best-Effort Application Identification of DPI-Serviced Flows” on page 18](#).



NOTE: For rms- interfaces, the statistics received from the active Multiservices PICs in the RMS group are combined with the statistics of the reported ended flows kept on the Routing Engine. The aggregated value is written to the statistics file. In the case of AMS interfaces, all the Multiservices PICs consisting of the AMS group reports statistics independently. These statistics are aggregated on the Routing Engine. The Routing Engine runs an independent timer, which on expiry writes the aggregated entry in the statistics file. This method of collection causes the statistics data in the statistics file to be displayed with a small delay.

Configuring an L-PDF Statistics Profile

You can specify an L-PDF statistics profile by including the following configuration at the **[edit accounting-options]** hierarchy level:

```
[edit accounting-options]
policy-decision-statistics-profile profile-name {
  application-aware-access-list-fields [ field-name ];
  file filename;
  files number;
  size bytes;
}
```



NOTE: This configuration method is not the preferred method for configuring Junos Application Aware (previously known as Dynamic Application Awareness) statistics. It is only maintained for backwards compatibility and may be deprecated in a future software release and does not support the use of IPv6 address and prefix length. The new, preferred configuration is found at the **[edit system services local-policy-decision-function]** hierarchy level, as described in “[Configuring an ACL Statistics Profile](#)” on page 64. We encourage you to migrate to the new configuration method.

You specify a profile name to identify the profile and other properties as needed by including the **policy-decision-statistics-profile** statement. The **aacl-fields** statement specifies which statistics to collect in an accounting-data log file. This log file is located on the **/var/log** directory on the router. You specify the log file by including the **file filename** statement. The filename is prefixed by the **aacl_statistics_** prefix; for example, if you specify the filename **lpdfd**, the log file will be **/var/log/aacl_statistics_lpdfd**.

The **application-aware-access-list-fields** statement supports the following options:

- **address**—IP Address
- **application**—Application name
- **application-group**—Application group name
- **input-bytes**—Number of input bytes
- **input-interface**—Input interface name
- **input-packets**—Number of input packets
- **mask**—Netmask
- **output-bytes**—Number of output bytes
- **output-packets**—Number of output packets
- **subscriber-name**—Subscriber name
- **timestamp**—Timestamp
- **vrf-name**—VPN routing and forwarding (VRF) name

For more information on configuring profiles, see the *Network Management Administration Guide for Routing Devices*.

Configuring an ACL Statistics Profile

You can specify an ACL statistics profile by including the following configuration at the **[edit system services]** hierarchy level:

```
local-policy-decision-function {
  statistics {
    file filename {
      archive-sites [ url ];
      files number;
      size bytes;
      transfer-interval minutes;
    }
    aacl-statistics-profile profile-name {
      aacl-fields [ field-name ];
      file filename;
      report-interval minutes;
      record-mode (interim-active-only | interim-full);
    }
    record-type (delta | interim);
  }
}
```

To specify the file properties, include the **file** statement at the **[edit system services local-policy-decision-function statistics]** hierarchy level with a unique filename:

- The **archive-sites** statement specifies one or more URLs for archiving the files. Archiving can be done by using FTP or SCP.
- The **files** statement specifies the maximum number of files that are maintained at one time.
- The **size** statement specifies the maximum size of each file.
- The **transfer-interval** statement specifies the interval between data transfers in minutes.

You specify a profile name to identify the profile and other properties as needed by including the **aacl-statistics-profile** statement. The **aacl-fields** statement specifies which statistics to collect in an accounting-data log file. This log file is located on the **/var/stats/aacl** directory on the router. You specify the log file by including the **file filename** statement.

The **aacl-fields** statement supports the following options:

- **address**—IP Address
- **all-fields**—All available fields
- **application**—Application name
- **application-group**—Application group name
- **input-bytes**—Number of input bytes
- **input-interface**—Input interface name

- **ipv6-address**—IPv6 address
- **ipv6-prefix-length**—Prefix length associated with the displayed IPv6 address
- **input-packets**—Number of input packets
- **mask**—Netmask
- **output-bytes**—Number of output bytes
- **output-packets**—Number of output packets
- **subscriber-name**—Subscriber name
- **timestamp**—Timestamp
- **vrf-name**—VPN routing and forwarding (VRF) name

The **record-type** statement specifies whether a record is **delta** or **interim**; **delta** is the default setting. The **report-interval** statement specifies the reporting interval in minutes; the default setting is 15 minutes and the range is 5 through 1440 minutes. The **record-mode** statement specifies how the statistics are reported for each reporting interval; the default setting is **interim-full** and reports all available statistics. To report only statistics that have changed for the reporting interval, use the **interim-active-only** setting.



NOTE: The IPv6 fields (**ipv6-address** and **ipv6-prefix-length**) are not supported for **record-type delta**. The IPv6 fields are supported for **record-type interim** only, meaning that the fields are restricted to the S- (Login) record.

For more information on configuring profiles, see the *Network Management Administration Guide for Routing Devices*.

Related Documentation

- [L-PDF Overview on page 57](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 18](#)
- [Applying L-PDF Profiles to Service Sets on page 65](#)
- [Tracing L-PDF Operations on page 67](#)

Applying L-PDF Profiles to Service Sets

You can optionally apply policy decision statistics profiles as part of a service-set definition. To do this, you include the **policy-decision-statistics-profile** statement at the **[edit services service-set *service-set-name*]** hierarchy level:

```
policy-decision-statistics-profile profile-name;
```



NOTE: To provide high availability for the policy decision statistics, associate the service-set definition with a redundant services PIC (rsp) interface.

You can include only one profile name in the specification for the **application-aware access-list** statement.

The following example shows a sample configuration for attachment of an L-PDF statistics profile:

```
services {
  service-set test_aacl_sset {
    aacl-rules aacl_rule;
    policy-decision-statistics-profile {
      pdf_stats_prof;
    }
    interface-service {
      service-interface ms-0/3/0.0;
    }
  }
}
```



NOTE: Only one service set can be applied to a single interface when L-PDF functionality is used.

The following example shows a sample configuration for attachment of a service set to a static interface:

```
interfaces {
  fe-0/0/0 {
    vlan-tagging;
    unit 1 {
      vlan-id 1;
      family inet {
        service {
          input {
            service-set test_aacl_sset;
          }
          output {
            service-set test_aacl_sset;
          }
        }
      }
      address 10.1.1.1/24;
    }
  }
}
```



NOTE: The `session-offload` statement at the `[edit chassis fpc slot-number pic number adaptive-services service-package extension-provider]` hierarchy level controls session offload behavior for Multiservices DPCs on MX Series routers. It controls session offload on a per-device basis, where a device is a Multiservices interface (`ms-fpc-pic-port`). Currently, the session offload function is supported for at most one Multiservices interface. When the offload function is enabled, it is strongly recommended that you limit Junos Application Aware (previously known as Dynamic Application Awareness) features to that Multiservices interface.

The default is to not offload any sessions. For more information on chassis configuration, see the *Junos OS Administration Library for Routing Devices*.

Related Documentation

- [L-PDF Overview on page 57](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 18](#)
- [Configuring Statistics Profiles on page 62](#)
- [Tracing L-PDF Operations on page 67](#)

Tracing L-PDF Operations

Tracing operations track L-PDF operations and record them in a log file. The logged error descriptions provide detailed information to help you solve problems faster.

By default, no events are traced. If you include the `traceoptions` statement at the `[edit system services local-policy-decision-function]` hierarchy level, you can customize the trace file settings:

```
traceoptions {
  file filename <files number> <size size>;
  flag flag;
}
```

The flags track the following information:

- **all**—Everything
- **configuration**—Configuration traces
- **database**—Database traces
- **general**—Miscellaneous traces
- **gres**—Graceful Routing Engine switchover (GRES) traces
- **ptsp-statistics**—PTSP statistics traces
- **rtsock**—Routing socket traces
- **statistics**—Statistics traces
- **subscriber**—Subscriber traces

- Related Documentation**
- [L-PDF Overview on page 57](#)
 - [Best-Effort Application Identification of DPI-Serviced Flows on page 18](#)
 - [Configuring Statistics Profiles on page 62](#)
 - [Applying L-PDF Profiles to Service Sets on page 65](#)

CHAPTER 5

Configuration Statements

- [\[edit services aacl\] Hierarchy List on page 71](#)
- [\[edit services application-identification\] Hierarchy Level on page 72](#)
- [aacl-fields on page 75](#)
- [aacl-statistics-profile on page 76](#)
- [address on page 77](#)
- [application \(Defining\) on page 78](#)
- [application \(Including in Rule\) on page 79](#)
- [application-aware-access-list-fields on page 80](#)
- [application-group on page 81](#)
- [application-group-any on page 81](#)
- [application-groups \(Services AACL\) on page 82](#)
- [application-groups \(Services Application Identification\) on page 82](#)
- [application-system-cache-timeout on page 83](#)
- [application-unknown on page 83](#)
- [applications \(Services AACL\) on page 83](#)
- [applications \(Services Application Identification\) on page 84](#)
- [automatic on page 84](#)
- [bypass-traffic-on-exceeding-flow-limits on page 85](#)
- [chain-order on page 85](#)
- [context on page 86](#)
- [destination \(Services\) on page 86](#)
- [destination-address on page 87](#)
- [destination-address-range on page 87](#)
- [destination-prefix-list on page 88](#)
- [direction on page 88](#)
- [disable \(APPID Application\) on page 89](#)
- [disable \(APPID Application Group\) on page 89](#)
- [disable \(APPID Port Mapping\) on page 89](#)

- [disable-global-timeout-override](#) on page 90
- [download](#) on page 90
- [enable-asymmetric-traffic-processing](#) on page 91
- [enable-heuristics](#) on page 91
- [file](#) on page 92
- [from](#) on page 93
- [idle-timeout](#) on page 94
- [ignore-errors](#) on page 94
- [index \(Applications\)](#) on page 95
- [index \(Nested Applications\)](#) on page 95
- [inactivity-non-tcp-timeout](#) on page 96
- [inactivity-tcp-timeout](#) on page 96
- [ip](#) on page 97
- [local-policy-decision-function](#) on page 98
- [log \(aACL\)](#) on page 99
- [match-direction](#) on page 99
- [max-checked-bytes](#) on page 100
- [maximum-transactions](#) on page 100
- [member](#) on page 101
- [min-checked-bytes](#) on page 101
- [nested-application](#) on page 102
- [nested-applications](#) on page 103
- [nested-application-settings](#) on page 103
- [nested-application-unknown](#) on page 104
- [no-application-identification](#) on page 104
- [no-application-system-cache](#) on page 105
- [no-clear-application-system-cache](#) on page 105
- [no-nested-application](#) on page 106
- [no-protocol-method](#) on page 106
- [no-signature-based](#) on page 107
- [order \(Services Application Identification\)](#) on page 107
- [pattern](#) on page 108
- [policy-decision-statistics-profile](#) on page 109
- [port-mapping](#) on page 110
- [port-range](#) on page 110
- [profile](#) on page 111
- [protocol](#) on page 111

- [rule \(ACL Rule Set\) on page 112](#)
- [rule \(Application Identification\) on page 113](#)
- [rule \(Including in Rule Set\) on page 114](#)
- [rule-set \(Services ACL\) on page 114](#)
- [rule-set \(Services Application Identification\) on page 115](#)
- [service-set-options on page 116](#)
- [statistics \(System Services\) on page 117](#)
- [support-uni-directional-traffic on page 117](#)
- [service-set \(Services\) on page 118](#)
- [services \(ACL\) on page 120](#)
- [services \(Application Identification\) on page 120](#)
- [session-timeout \(Application Identification\) on page 121](#)
- [session-timeout \(Interfaces\) on page 121](#)
- [signature on page 122](#)
- [signature-method-all-ports on page 122](#)
- [source on page 123](#)
- [source-address \(ACL\) on page 123](#)
- [source-address-range on page 124](#)
- [source-prefix-list \(Services ACL\) on page 124](#)
- [source-prefix-list \(Services IDS\) on page 125](#)
- [term on page 126](#)
- [then on page 127](#)
- [traceoptions \(Application Identification\) on page 129](#)
- [traceoptions \(Services Local Policy Decision Function\) on page 131](#)
- [type on page 132](#)
- [type-of-service on page 133](#)
- [url on page 133](#)

[edit services aacl] Hierarchy List

To configure application-aware access list (ACL) services, include the **aacl** statements at the **[edit services]** hierarchy level:

```
[edit services]
aacl {
  rule rule-name {
    match-direction (input | output | input-output);
    term term-name {
      from {
        application-group-any;
        application-groups [ application-group-names ];
        application-unknown;
```

```
    applications [ application-names ];
    destination-address address <any-unicast>;
    destination-address-range low minimum-value high maximum-value;
    destination-prefix-list list-name;
    source-address address <any-unicast>;
    source-address-range low minimum-value high maximum-value;
    source-prefix-list list-name;
  }
  then {
    (accept | discard);
    count (application | application-group | application-group-any | none);
    forwarding-class class-name;
    policer policer-name;
  }
}
}
rule-set rule-set-name {
  [ rule rule-names ];
}
}
```

**Related
Documentation**

- [Configuring ACL Rules on page 21](#)
- [Configuring ACL Rule Sets on page 26](#)
- [Configuring Logging of ACL Flows on page 27](#)

[edit services application-identification] Hierarchy Level

To configure application identification services (APPID), include the **application-identification** statement at the **[edit services]** hierarchy level:

```
[edit services]
application-identification {
  application application-name {
    disable;
    idle-timeout seconds;
    index number;
    session-timeout seconds;
    type type;
    type-of-service service-type;
    port-mapping {
      port-range {
        tcp (port | range);
        udp (port | range);
      }
      disable;
    }
  }
}
application-group group-name {
  application-groups {
    name [application-group-name];
  }
  applications {
    name [application-name];
  }
}
```



```

    }
    index number;
    disable;
  }
  application-system-cache-timeout seconds;
  enable-heuristics
  max-checked-bytes bytes;
  min-checked-bytes bytes;
  nested-application
  nested-application-settings
  no-application-identification;
  no-application-system-cache;
  no-clear-application-system-cache;
  no-protocol-method;
  no-signature-based;
  profile profile-name {
    [ rule-set rule-set-name ];
  }
  rule rule-name {
    disable;
    address address-name {
      destination {
        ip address </prefix-length>;
        port-range {
          tcp [ ports-and-port-ranges ];
          udp [ ports-and-port-ranges ];
        }
      }
      source {
        ip address </prefix-length>;
        port-range {
          tcp [ ports-and-port-ranges ];
          udp [ ports-and-port-ranges ];
        }
      }
      order number;
    }
    application application-name;
  }
  rule-set rule-set-name {
    rule application-rule-name;
  }
  signature-method-all-ports
  traceoptions {
    file filename <files number> <match regex> <size size> <world-readable |
      no-world-readable>;
    flag flag;
    no-remote-trace;
  }
}

[edit services]
hcm {
  url-rule url-rule-name {
    term term-num {
      from {

```

```
        url-list url-list-name ;
        url url_identifier {
            host hostname ;
            request-url page-name ;
        }
    }
    then {
        discard;
        accept;
        count;
        log-request;
    }
}
}
url-rule-set url-rule-set-name {
    url-rule rule1 ;
    url-rule rule2 ;
}
}
```

**Related
Documentation**

- [Defining an Application Identification on page 34](#)
- [Application Identification for Nested Applications on page 41](#)
- [Configuring Global APPID Properties on page 43](#)

aac1-fields

Syntax	<pre>aac1-fields { field-name; }</pre>
Hierarchy Level	[edit system services local-policy-decision-function statistics aac1-statistics-profile <i>profile-name</i>]
Release Information	Statement introduced in Junos OS Release 10.0. IPv6 support introduced in Junos OS Release 12.2.
Description	Define the statistics to collect in a data log file.
Options	<p><i>field-name</i>—Name of the field:</p> <ul style="list-style-type: none"> • address—IPv4 address • all-fields—All available fields • application—Application name • application-group—Application group name • input-bytes—Number of input bytes • input-interface—Input interface name • input-packets—Number of input packets • ipv6-address—IPv6 address • ipv6-prefix-length—Prefix length associated with the displayed IPv6 address • mask—Netmask • output-bytes—Number of output bytes • output-packets—Number of output packets • subscriber-name—Subscriber name • timestamp—Timestamp • vrf-name—VPN routing and forwarding (VRF) name
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Statistics Profiles on page 62

aacl-statistics-profile

Syntax	<pre>aacl-statistics-profile <i>profile-name</i> { aacl-fields { <i>field-name</i>; } file <i>filename</i>; record-mode (interim-active-only interim-full); report-interval <i>minutes</i>; }</pre>
Hierarchy Level	[edit services service-set <i>service-set-name</i>], [edit system services local-policy-decision-function statistics]
Release Information	Statement introduced in Junos OS Release 10.0. record-mode option introduced in Junos OS Release 10.2.
Description	Create an ACL statistics profile, which configures the files to which statistics records are exported and the format that is exported.
Options	<p>file <i>filename</i>—Name of the file to receive the statistics data output. Enclose the name within quotation marks. All files are placed in the directory <code>/var/stats/aacl</code>.</p> <p>record-mode—Record mode for the reporting interval; possible values are interim-active-only, which reports only statistics that have changed, or interim-full, which reports all available statistics.</p> <p>report-interval <i>minutes</i>—Frequency at which statistics are recorded, in minutes.</p> <p>Default: 15 minutes</p> <p>Range: 5 through 1440 minutes</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface —To view this statement in the configuration. interface-control —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">For more information on profiles, see the <i>Network Management Administration Guide for Routing Devices</i>.Configuring Statistics Profiles on page 62

address

```
Syntax  address address-name {
          destination {
            ip address</prefix-length>;
            port-range {
              tcp [ ports-and-port-ranges ];
              udp [ ports-and-port-ranges ];
            }
          }
          source {
            ip address</prefix-length>;
            port-range {
              tcp [ ports-and-port-ranges ];
              udp [ ports-and-port-ranges ];
            }
          }
          order number;
        }
```

Hierarchy Level [edit services application-identification **rule** *rule-name*]

Release Information Statement introduced in Junos OS Release 9.5.

Description Define address properties for application-identification rule processing. This statement is mandatory; you must specify either the destination or source properties.

Options *address-name*—Identifier for address information.

The remaining statements are explained separately.

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

Related Documentation

- [Configuring APPID Rules on page 36](#)

application (Defining)

Syntax `application application-name {
 disable;
 idle-timeout seconds;
 index number;
 port-mapping {
 disable;
 port-range {
 tcp [ports-and-port-ranges];
 udp [ports-and-port-ranges];
 }
 }
 session-timeout seconds;
 type type;
 type-of-service service-type;
 }`

Hierarchy Level [edit services application-identification]

Release Information Statement introduced in Junos OS Release 9.5.

Description Define the application and its properties.

The remaining statements are explained separately.

Options *application-name*—Identifier for the application. This is a mandatory value and has a maximum length of 32 characters.

Required Privilege interface—To view this statement in the configuration.

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

Related Documentation

- [Defining an Application Identification on page 34](#)

application (Including in Rule)

Syntax	<code>application <i>application-name</i>;</code>
Hierarchy Level	[edit services application-identification rule <i>rule-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify the application for inclusion in a rule.
Options	<i>application-name</i> —Identifier for the application.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring APPID Rules on page 36

application-aware-access-list-fields

Syntax	<pre>application-aware-access-list-fields { <i>field-name</i>; }</pre>
Hierarchy Level	[edit accounting-options policy-decision-statistics-profile <i>profile-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define the statistics to collect in a data log file.
Options	<p><i>field-name</i>—Name of the field:</p> <ul style="list-style-type: none">• address—IP address• application—Application name• application-group—Application group name• input-bytes—Number of input bytes• input-interface—Input interface name• input-packets—Number of input packets• mask—Netmask• output-bytes—Number of output bytes• output-packets—Number of output packets• subscriber-name—Subscriber name• timestamp—Timestamp• vrf-name—VPN routing and forwarding (VRF) name
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Statistics Profiles on page 62

application-group

Syntax	<pre> application-group <i>group-name</i> { disable; application-groups { <i>application-group-name</i>; } applications { <i>application-name</i>; } index <i>number</i>; } </pre>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define the properties and contents of the application group.
Options	<p><i>group-name</i>—Unique identifier for the group.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Application Groups on page 40

application-group-any

Syntax	application-group-any;
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Match any application group defined in the database.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in AACL Rules on page 22

application-groups (Services ACL)

Syntax	<code>application-groups [<i>application-group-names</i>];</code>
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify one or more application groups defined in the application identification configuration for inclusion as a match condition.
Options	<i>application-group-names</i> —Identifiers of the application groups.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Match Conditions in ACL Rules on page 22

application-groups (Services Application Identification)

Syntax	<code>application-groups { <i>application-group-name</i>; }</code>
Hierarchy Level	[edit services application-identification application-group <i>group-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify the list of application groups for inclusion in a larger application group. An <i>application-group-name</i> statement is mandatory.
Options	<i>application-group-name</i> —Identifier for the application group. Maximum length is 32 characters.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Application Groups on page 40

application-system-cache-timeout

Syntax	<code>application-system-cache-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Configure the lifetime for entries in the application system cache.
Options	<i>seconds</i> —Lifetime for system cache entries, in seconds.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Global APPID Properties on page 43

application-unknown

Syntax	<code>application-unknown</code>
Hierarchy Level	[edit services aacl <i>rule rule-name term term-name from</i>]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enable AACL logging of flows for unknown applications.
Related Documentation	<ul style="list-style-type: none"> • See Configuring Logging of AACL Flows on page 27.

applications (Services AACL)

Syntax	<code>applications [<i>application-names</i>];</code>
Hierarchy Level	[edit services aacl <i>rule rule-name term term-name from</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify one or more applications defined in the application identification configuration for inclusion as a match condition.
Options	<i>application-names</i> —Identifiers of the applications.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in AACL Rules on page 22

applications (Services Application Identification)

Syntax	<pre>applications { <i>application-name</i>; }</pre>
Hierarchy Level	[edit services application-identification application-group group-name]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify the list of applications for inclusion in the application group.
Options	<i>application-name</i> —Identifier for the application. Maximum length is 32 characters.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Application Groups on page 40

automatic

Syntax	<pre>automatic { interval <i>hour</i>; start-time <i>time</i>; }</pre>
Hierarchy Level	[edit services application-identification download]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define automatic download properties.
Options	<i>interval hour</i> —Download interval in hours. The default is 24 and the range is from 1 through 168. <i>start-time time</i> —Start-time value. The default is 0:00 and the range is from 0:00 through 24:00.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Automatic Download of Application Package Updates on page 46

bypass-traffic-on-exceeding-flow-limits

Syntax	bypass-traffic-on-exceeding-flow-limits;
Hierarchy Level	[edit services service-set <i>service-set-name</i> service-set-options]
Release Information	Statement introduced in Junos OS Release 10.1.
Description	Enable packets to bypass without creating a new session when the flow in the service set exceeds the limit that is set by the max-flows statement at the [edit services service-set <i>service-set-name</i>] hierarchy level.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Configuring Service Sets to be Applied to Services Interfaces</i>

chain-order

Syntax	chain-order;
Hierarchy Level	[edit services application-identification <i>nested-application name signature name</i>]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Signatures can contain multiple members. If the chain order feature is on, those members are read in order. By default, chain ordering is turned off. If a signature contains only one member, this option is ignored.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Application Identification for Nested Applications on page 41

context

Syntax	context (http-header-content-type http-header-host http-url-parsed http-url-parsed-param-parsed);
Hierarchy Level	[edit services application-identification nested-application name signature name member name]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Define a service-specific context, such as http-url .
Options	value —Service-specific context.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

destination (Services)

Syntax	<pre>destination { ip address</prefix-length>; port-range { tcp [<i>ports-and-port-ranges</i>]; udp [<i>ports-and-port-ranges</i>]; } }</pre>
Hierarchy Level	[edit services application-identification rule rule-name address address-name]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define destination properties for application-identification rule processing.
Options	The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring APPID Rules on page 36

destination-address

Syntax	<code>destination-address <i>address</i>;</code>
Hierarchy Level	[edit services aacl rule rule-name term term-name from]
Release Information	Statement introduced in Junos OS Release 9.5. IPv6 support introduced in Junos OS Release 12.2.
Description	Specify the destination address for rule matching.
Options	<i>address</i> —Destination IPv4 or IPv6 address or prefix value.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in AACL Rules on page 22

destination-address-range

Syntax	<code>destination-address-range low <i>minimum-value</i> high <i>maximum-value</i>;</code>
Hierarchy Level	[edit services aacl rule rule-name term term-name from]
Release Information	Statement introduced in Junos OS Release 9.5. IPv6 support introduced in Junos OS Release 12.2.
Description	Specify the destination address range for rule matching.
Options	<i>minimum-value</i> —Lower boundary for the IPv4 or IPv6 address range. <i>maximum-value</i> —Upper boundary for the IPv4 or IPv6 address range.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in AACL Rules on page 22

destination-prefix-list

Syntax	<code>destination-prefix-list <i>list-name</i>;</code>
Hierarchy Level	<code>[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]</code>
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the destination prefix list for rule matching. You configure the prefix list by including the prefix-list statement at the [edit policy-options] hierarchy level.
Options	<i>list-name</i> —Destination prefix list.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Match Conditions in ACL Rules on page 22

direction

Syntax	<code>direction (any client-to-server server-to-client) ;</code>
Hierarchy Level	<code>[edit services application-identification nested-application <i>name</i> signature <i>name</i> member <i>name</i>]</code>
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Specify the connection direction of the packets to apply pattern matching.
Options	<i>direction</i> —The directions of packets are client-to-server , server-to-client , or any .
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41.

disable (APPID Application)

Syntax	disable;
Hierarchy Level	[edit services application-identification application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable this application definition.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34

disable (APPID Application Group)

Syntax	disable;
Hierarchy Level	[edit services application-identification application-group <i>group-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable application group properties.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Application Groups on page 40

disable (APPID Port Mapping)

Syntax	disable;
Hierarchy Level	[edit services application-identification application <i>application-name</i> port-mapping]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable port-mapping properties for application identification.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34

disable-global-timeout-override

Syntax	disable-global-timeout-override;
Hierarchy Level	[edit interfaces <i>interface-name</i> services-options]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Disallow overriding a global inactivity or session timeout.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34

download

Syntax	<pre>download { automatic { interval <i>hour</i>; start-time <i>time</i>; } url <i>url</i>; }</pre>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define application download properties.
Options	The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Automatic Download of Application Package Updates on page 46

enable-asymmetric-traffic-processing

Syntax	enable-asymmetric-traffic-processing;
Hierarchy Level	[edit services service-set <i>service-set-name</i> service-set-options]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enable APPID to perform application matching on unidirectional traffic.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring APPID Support for Unidirectional Traffic on page 45

enable-heuristics

Syntax	enable-heuristics;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enable APPID to identify encrypted data packets in point-to-point applications by using heuristics methodology.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring APPID Support for Heuristics on page 44

file

Syntax	<pre>file <i>file-name</i> { archive-sites <i>url</i>; files <i>file-number</i>; size <i>bytes</i>; transfer-interval <i>minutes</i>; }</pre>
Hierarchy Level	[edit system services local-policy-decision-function statistics]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Specify a file to which statistics records are exported and the format that is exported.
Options	<p>archive-sites [<i>url</i>]—One or more destinations for archiving data.</p> <p><i>file-name</i>—Name of the file to receive the statistics data output.</p> <p>files <i>file-number</i>—(Optional) Maximum number of accounting files.</p> <p>Range: 3 through 1000 files</p> <p>Default: 3 files</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option.</p> <p>size <i>bytes</i>—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB).</p> <p>Syntax: <i>xk</i> to specify KB, <i>xm</i> to specify MB, or <i>xg</i> to specify GB</p> <p>Range: 262144 through 1073741824 or the maximum file size supported on your system</p> <p>If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.</p> <p>transfer-interval <i>minutes</i>—Frequency at which to transfer files to archive sites, in minutes.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Statistics Profiles on page 62


from

Syntax	<pre> from { application-group-any; application-groups [application-group-names]; application-unknown; applications [application-names]; destination-address address <any-unicast>; destination-address-range low minimum-value high maximum-value; destination-prefix-list list-name; nested-application-unknown; source-address address <any-unicast>; source-address-range low minimum-value high maximum-value; source-prefix-list list-name; } </pre>
Hierarchy Level	[edit services aacl rule rule-name term term-name]
Release Information	Statement introduced before Junos OS Release 9.5.
Description	Specify match conditions for the ACL term.
Options	<p>For information on match conditions, see the description of firewall filter match conditions in the <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i>.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring ACL Rules on page 21

idle-timeout

Syntax	<code>idle-timeout seconds;</code>
Hierarchy Level	[edit services application-identification application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define idle timeout for an application in seconds. When the timeout period expires, the session ends if no packets have been received.
Options	seconds —Idle timeout period. Default: 30 Range: 1 through 604,800
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• APPID Overview on page 29• Defining an Application Identification on page 34

ignore-errors

Syntax	<code>ignore-errors <alg> <tcp>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> services-options]
Release Information	Statement introduced in Junos OS Release 10.1.
Description	Define settings for minimizing TCP packet drops during stateful firewall processing.
<div> NOTE: ignore-errors option is not supported on adaptive services interfaces (sp-x/y/z).</div>	
Options	alg —(Optional) Mediate ALG behavior that results in dropping malformed packets or random packets when the software is unable to allocate resources. tcp —(Optional) Prevent software from dropping packets that fail TCP SYN checks.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34

index (Applications)

Syntax	<code>index number;</code>
Hierarchy Level	[edit services application-identification application application-name], [edit services application-identification application-group group-name]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Assign an application or application-group index number. This is a mandatory value.
Options	number —Index number; must be a unique, unsigned value. Range: 0 through 65535
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34 • Configuring Application Groups on page 40

index (Nested Applications)

Syntax	<code>index number;</code>
Hierarchy Level	[edit services application-identification nested-application name]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Set a number that is a one-to-one mapping to the application name. The application name is used to ensure that each signature definition is unique.
Options	number —Numeric value associated with an application name. The index range for predefined applications is from 1 through 32767. The index range for custom applications and custom nested applications is from 32768 through 65534.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Application Identification for Nested Applications on page 41.

inactivity-non-tcp-timeout

Syntax	<code>inactivity-non-tcp-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> services-options]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Define the inactivity timeout period for non-TCP established sessions in seconds.
Options	<i>seconds</i> —Timeout period. Range: 4 through 86,400
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34

inactivity-tcp-timeout

Syntax	<code>inactivity-tcp-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> services-options]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Define the inactivity timeout period for TCP established sessions in seconds.
Options	<i>seconds</i> —Timeout period. Range: 4 through 86,400
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34

ip

Syntax	<code>ip address</prefix-length>;</code>
Hierarchy Level	[edit services application-identification rule rule-name address destination], [edit services application-identification rule rule-name address source]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define an IP address and netmask for identifying the traffic destination or source.
Options	<code>address</prefix-length></code> —IP address and netmask.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring APPID Rules on page 36

local-policy-decision-function

Syntax local-policy-decision-function {
 statistics {
 aocl-statistics-profile *profile-name* {
 aocl-fields {
 field-name;
 }
 file *filename*;
 report-interval *minutes*;
 }
 file *file-name* {
 archive-sites *url*;
 files *file-number*;
 size *bytes*;
 transfer-interval *minutes*;
 }
 record-type (delta | interim);
 }
 traceoptions {
 file *filename* <files *number*> <size *size*>;
 flag *flag*;
 no-remote-trace;
 }
 }

Hierarchy Level [edit system services]

Release Information Statement introduced in Junos OS Release 10.0.

Description Specify L-PDF properties.

Options The remaining statements are explained separately.

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

Related Documentation • [Configuring Statistics Profiles on page 62](#)

log (aACL)

Syntax	log <i>event-type</i>
Hierarchy Level	[edit services aACL rule <i>rule-name</i> term <i>term-name</i> then]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enable AACL logging of flows for known or unknown applications.
Options	<i>event-type</i> —Enable logging of the specified <i>event-type</i> : <ul style="list-style-type: none"> • session-start • session-end • session-start-end-no-stats • session-start-interim-end • session-interim end • session-end
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • See Configuring Logging of AACL Flows on page 27.

match-direction

Syntax	match-direction (input output input-output);
Hierarchy Level	[edit services aACL rule <i>rule-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the direction in which the rule match is applied.
Options	<i>input</i> —Apply the rule match on the input side of the interface. <i>output</i> —Apply the rule match on the output side of the interface. <i>input-output</i> —Apply the rule match bidirectionally.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Direction for AACL Rules on page 22

max-checked-bytes

Syntax	<code>max-checked-bytes <i>bytes</i>;</code>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the maximum number of bytes to be inspected.
Options	<i>bytes</i> —Maximum number of bytes. Range: 0 through 100,000
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Global APPID Properties on page 43

maximum-transactions

Syntax	<code>maximum-transactions <i>number</i>;</code>
Hierarchy Level	[edit services application-identification nested-application name signature name]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Set the maximum number of transactions required before a match is made.
Options	<i>number</i> —Maximum number of transactions.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

member

Syntax	<code>member <i>name</i>;</code>
Hierarchy Level	[edit services application-identification nested-application <i>name</i> signature <i>name</i>]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Define a member name for a custom nested application signature definition. Custom definitions can contain multiple members that define attributes for an application.
Options	<i>name</i> —Name of member for a custom nested application signature definition.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Application Identification for Nested Applications on page 41

min-checked-bytes

Syntax	<code>min-checked-bytes <i>bytes</i>;</code>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the minimum number of bytes to be inspected.
Options	<i>bytes</i> —Minimum number of bytes. Range: 0 through 2000
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Global APPID Properties on page 43

nested-application

Syntax	<pre>nested-application <i>name</i> { <i>index</i> <i>number</i>; <i>protocol</i> <i>protocol</i> ; <i>signature</i> <i>name</i> { <i>chain-order</i> ; <i>maximum-transactions</i> <i>number</i>; <i>member</i> <i>name</i> { <i>context</i> (http-header-content-type http-header-host http-url-parsed http-url-parsed-param-parsed); <i>direction</i> (any client-to-server server-to-client); <i>pattern</i> <i>dfa-pattern</i>; } <i>order</i> <i>number</i>; } <i>type</i> <i>type</i>; }</pre>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Configure a custom nested application definition for the desired application name that will be used by the system to identify the nested application as it passes through the device. Custom nested application definitions can be used for nested applications that are not part of the Juniper Networks predefined nested application database.
Options	<p><i>name</i>—Name of nested application.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

nested-applications

Syntax	<code>nested-applications [<i>nested-application-names</i>];</code>
Hierarchy Level	<code>[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]</code>
Release Information	Statement introduced in Junos OS Release 11.1.
Description	Identify one or more nested applications defined in the application identification configuration for inclusion as a match condition.
Options	<i>nested-application-names</i> —Identifiers of the nested applications.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in ACL Rules on page 22

nested-application-settings

Syntax	<pre>nested-application-settings { no-application-system-cache; no-nested-application; }</pre>
Hierarchy Level	<code>[edit services application-identification]</code>
Release Information	Statement introduced in Junos OS Release 10.2.
Description	<p>Configure nested application options for application identification services.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Application Identification for Nested Applications on page 41.

nested-application-unknown

Syntax	nested-application-unknown
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 11.4.
Description	Enable AACL logging of flows for unknown nested applications.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Logging of AACL Flows on page 27.

no-application-identification

Syntax	no-application-identification;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable all application identification methods.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Global APPID Properties on page 43

no-application-system-cache

Syntax	no-application-system-cache;
Hierarchy Level	[edit services application-identification], [edit services application-identification nested-application-settings]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable storing application identification results in the application system cache. Nested application identification information is saved in the application system cache to improve performance. This cache is updated when a different application is identified. This caching is turned on by default. Use the no-application-system-cache statement to turn it off.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Global APPID Properties on page 43 • Application Identification for Nested Applications on page 41.

no-clear-application-system-cache

Syntax	no-clear-application-system-cache;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable clearing the application system cache.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Global APPID Properties on page 43

no-nested-application

Syntax	no-nested-application;
Hierarchy Level	[edit services application-identification nested-application-settings]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Sometimes there is a need to identify multiple different applications running on the same Layer 7 protocols. For example, both Facebook and Yahoo Messenger can run over HTTP, but there is a need to identify them as two different applications. To do this, the application layer is split into two layers: Layer 7 applications and Layer 7 protocols. This function is turned on by default. Use the no-nested-application statement to turn it off.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

no-protocol-method

Syntax	no-protocol-method;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 10.1.
Description	Disable the protocol-based application identification method.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Global APPID Properties on page 43

no-signature-based

Syntax	no-signature-based;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Disable the signature-based application identification method.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Global APPID Properties on page 43

order (Services Application Identification)

Syntax	order <i>number</i> ;
Hierarchy Level	[edit services application-identification <i>nested-application name signature name member name</i>] [edit services application-identification <i>rule rule-name address</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define application matching priority. For address configurations, the order number resolves the conflict when multiple address entries are matched for a specific session. The lower number has higher priority.
Options	<i>number</i> —Order number. This value is mandatory and must be unique.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring APPID Rules on page 36 • Application Identification for Nested Applications on page 41

pattern

Syntax	<code>pattern <i>dfa-pattern</i>;</code>
Hierarchy Level	[edit services application-identification nested-application name signature name member name]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Define an attack pattern to be detected.
Options	<i>dfa-pattern</i> —Pattern of attack to match. Deterministic Finite Automata (DFA) is a powerful pattern matching engine.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

policy-decision-statistics-profile

Syntax	<pre> policy-decision-statistics-profile <i>profile-name</i> { aacl-fields { <i>field-name</i>; } file <i>filename</i>; files <i>file-number</i>; size <i>bytes</i>; } </pre>
Hierarchy Level	[edit accounting-options], [edit services service-set <i>service-set-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Create a policy decision statistics profile, which configures the files to which statistics records are exported and the format that is exported.
Options	<p>file <i>filename</i>—Name of the file to receive the accounting-data output. Enclose the name within quotation marks. All files are placed in the directory <code>/var/log</code>.</p> <p>files <i>number</i>—(Optional) Maximum number of accounting files. Range: 2 through 1000 files Default: 2 files</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option.</p> <p><i>profile-name</i>—Name of the policy decision statistics profile.</p> <p>size <i>size</i>—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). Syntax: <i>xk</i> to specify KB, <i>xm</i> to specify MB, or <i>xg</i> to specify GB Range: 10240 through 1073741824 or the maximum file size supported on your system</p> <p>If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> For more information on profiles, see the <i>Network Management Administration Guide for Routing Devices</i>. Configuring Statistics Profiles on page 62

port-mapping

Syntax	<pre>port-mapping { disable; port-range { tcp [<i>ports-and-port-ranges</i>]; udp [<i>ports-and-port-ranges</i>]; } }</pre>
Hierarchy Level	[edit services application-identification application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define port-mapping properties for application identification.
Options	The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34

port-range

Syntax	<pre>port-range { tcp [<i>ports-and-port-ranges</i>]; udp [<i>ports-and-port-ranges</i>]; }</pre>
Hierarchy Level	[edit services application-identification application <i>application-name</i> port-mapping], [edit services application-identification rule <i>rule-name</i> address destination], [edit services application-identification rule <i>rule-name</i> address source]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define TCP and UDP port numbers or numeric ranges. For port-mapping configurations, this entry is required if the parent node exists.
Options	<i>ports-and-port-ranges</i> —Individual port numbers, numeric port ranges, or both. Separate the values with spaces. The format for numeric port ranges is <i>minimum-value–maximum-value</i> .
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34• Configuring APPID Rules on page 36

profile

Syntax	<code>profile <i>profile-name</i> { <i>rule-set</i> <i>rule-set-name</i>; }</code>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define members of application profile, which consists of one or more rule sets.
Options	<p><i>profile-name</i>—Identifier for application profile.</p> <p>The remaining statement is explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Application Profiles on page 39

protocol

Syntax	<code>protocol <i>protocol</i>;</code>
Hierarchy Level	[edit services application-identification <i>nested-application name</i>]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Identify the protocol that will be monitored to identify nested applications. HTTP is supported.
Options	<i>protocol</i> —An agreed-upon or standardized method for transmitting data and establishing communications between different devices. The value http is supported.
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Application Identification for Nested Applications on page 41

rule (AACL Rule Set)

Syntax	<pre> rule <i>rule-name</i> { match-direction (input output input-output); term <i>term-name</i> { from { application-group-any; application-groups [<i>application-group-names</i>]; application-unknown; applications [<i>application-names</i>]; destination-address <i>address</i> <any-unicast>; destination-address-range low <i>minimum-value</i> high <i>maximum-value</i>; destination-prefix-list <i>list-name</i>; nested-application-unknown; source-address <i>address</i> <any-unicast>; source-address-range low <i>minimum-value</i> high <i>maximum-value</i>; source-prefix-list <i>list-name</i>; } then { (accept discard); count (application application-group application-group-any nested-application none); forwarding-class <i>class-name</i>; policer <i>policer-name</i>; } } } </pre>
Hierarchy Level	[edit services aacl], [edit services aacl rule-set <i>rule-set-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the rule the router uses when applying this service.
Options	<p>rule-name—Identifier for the collection of terms that constitute this rule.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring AACL Rules on page 21

rule (Application Identification)

```
Syntax  rule rule-name {
        address {
            destination {
                ip address </prefix-length>;
                port-range {
                    tcp [ ports-and-port-ranges ];
                    udp [ ports-and-port-ranges ];
                }
            }
            source {
                ip address </prefix-length>;
                port-range {
                    tcp [ ports-and-port-ranges ];
                    udp [ ports-and-port-ranges ];
                }
            }
            order number;
        }
        application application-name;
    }
```

Hierarchy Level [edit services application-identification]

Release Information Statement introduced in Junos OS Release 9.5.

Description Define properties for application-identification rule processing.

Options *rule-name*—Unique identifier for the rule.

The remaining statements are explained separately.

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

Related Documentation

- [Configuring APPID Rules on page 36](#)

rule (Including in Rule Set)

Syntax	<code>rule rule-name;</code>
Hierarchy Level	[edit services application-identification rule-set rule-set-name]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Identify rules for inclusion in application rule set.
Options	<i>rule-name</i> —Unique identifier for the rule.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring APPID Rules on page 36

rule-set (Services ACL)

Syntax	<pre>rule-set rule-set-name { [rule rule-names]; }</pre>
Hierarchy Level	[edit services aacl]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the rule set the router uses when applying this service.
Options	<i>rule-set-name</i> —Identifier for the collection of rules that constitute this rule set. The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring AACL Rule Sets on page 26

rule-set (Services Application Identification)

Syntax	<code>rule-set <i>rule-set-name</i> { <code>rule</code> <i>application-rule-name</i>; }</code>
Hierarchy Level	[edit services application-identification], [edit services application-identification <code>profile</code> <i>profile-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define members of rule set.
Options	<i>rule-set-name</i> —Unique identifier for the rule set. The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring APPID Rules on page 36

service-set-options

Syntax	<pre>service-set-options { bypass-traffic-on-exceeding-flow-limits; bypass-traffic-on-pic-failure; enable-asymmetric-traffic-processing; header-integrity-check routing-engine-services; support-uni-directional-traffic; enable-change-on-ams-redistribution; }</pre>
Hierarchy Level	[edit services service-set]
Release Information	<p>Statement introduced in Junos OS Release 10.1.</p> <p>The enable-asymmetric-traffic-processing and the support-uni-directional-traffic options were added in Junos OS Release 11.2.</p> <p>The routing-engine-services option was added in Junos OS Release 15.1.</p> <p>enable-change-on-ams-redistribution option added in Junos OS Release 15.1</p>
Description	Specify the service set options to apply to a service set.
Options	The remaining statements are explained separately.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Service Sets to be Applied to Services Interfaces</i>• Configuring APPID Support for Unidirectional Traffic on page 45• <i>Enabling the Reset of Service Sets for Aggregated Multiservices Interfaces</i>

statistics (System Services)

Syntax	<pre> statistics { aacl-statistics-profile <i>profile-name</i> { aacl-fields { <i>field-name</i>; } file <i>filename</i>; report-interval <i>minutes</i>; } file <i>file-name</i> { archive-sites [<i>url</i>]; files <i>file-number</i>; size <i>bytes</i>; transfer-interval <i>minutes</i>; } record-type (delta interim); }</pre>
Hierarchy Level	[edit system services local-policy-decision-function]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Configure file and data specifications for recording ACL statistics.
Options	<p>record-type—Record type; possible values are delta or interim.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Statistics Profiles on page 62

support-uni-directional-traffic

Syntax	support-uni-directional-traffic;
Hierarchy Level	[edit services service-set <i>service-set-name</i> service-set-options]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enable APPID to perform application matching on unidirectional traffic.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring APPID Support for Unidirectional Traffic on page 45

service-set (Services)

```

Syntax  service-set service-set-name {
        allow-multicast;
        extension-service service-name {
            provider-specific-rules-configuration;
        }
        (ids-rules rule-name | ids-rule-sets rule-set-name);
        interface-service {
            service-interface interface-name;
        }
        ipsec-vpn-options {
            anti-replay-window-size bits;
            clear-dont-fragment-bit;
            ike-access-profile profile-name;
            local-gateway address;
            no-anti-replay;
            passive-mode-tunneling;
            trusted-ca [ ca-profile-names ];
            tunnel-mtu bytes;
        }
        ip-reassembly-rules rule-name;
        (ipsec-vpn-rules rule-name | ipsec-vpn-rule-sets rule-set-name);
        max-flows number;
        max-drop-flows {
            ingress ingress-flows;
            egress egress-flows;
        }
        nat-options {
            land-attack-check (ip-only | ip-port);
            max-sessions-per-subscriber session-number;
            stateful-nat64 {
                clear-dont-fragment-bit;
            }
        }
        (nat-rules rule-name | nat-rule-sets rule-set-name);
        next-hop-service {
            inside-service-interface interface-name.unit-number;
            outside-service-interface interface-name.unit-number;
            outside-service-interface-type local;
            service-interface-pool name;
        }
        (pgcp-rules rule-name | pgcp-rule-sets rule-set-name);
        (ptsp-rules rule-name | ptsp-rule-sets rule-set-name);
        service-set-options {
            bypass-traffic-on-exceeding-flow-limits;
            bypass-traffic-on-pic-failure;
            enable-asymmetric-traffic-processing;
            routing-engine-services;
            support-uni-directional-traffic;
        }
        snmp-trap-thresholds {
            flows high high-threshold | low low-threshold;
            nat-address-port high-threshold | low low-threshold;
        }
    }

```

```

    }
  }
  software-options {
    dslite-ipv6-prefix-length dslite-ipv6-prefix-length;
  }
  (software-rules rule-name | software-rule-sets rule-set-name);
  (stateful-firewall-rules rule-name | stateful-firewall-rule-sets rule-set-name);
  syslog {
    host hostname {
      class {
        alg-logs;
        ids-logs;
        nat-logs;
        packet-logs;
        pcp-logs;
        session-logs <open | close>;
        stateful-firewall-logs ;
      }
      services severity-level;
      facility-override facility-name;
      interface-service prefix-value;
    }
  }
}

```

Hierarchy Level [edit services]

Release Information Statement introduced before Junos OS Release 7.4.
pgcp-rules and **pgcp-rule-sets** options added in Junos OS Release 8.4.
server-set-options option added in Junos OS Release 10.1.
ptsp-rules and **ptsp-rule-sets** options added in Junos OS Release 10.2.
software-rules and **clear-rule-sets** options added in Junos OS Release 10.4.
software-options option added in Junos OS Release 14.1.

Description Define the service set.

Options ***service-set-name***—Name of the service set. You can include special characters, such as a forward slash (/), colon (:), or a period (.).

Range: Up to 64 alphanumeric characters.

The remaining statements are explained separately.

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

Related Documentation

- *Service Set Properties*

services (AACL)

Syntax	<code>services aacl { ... }</code>
Hierarchy Level	[edit]
Release Information	aacl statement introduced in Junos OS Release 9.5.
Description	Define the services to be applied to traffic.
Options	aacl —Use the values configured for application-aware-access-list matching rules.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Application Aware Services Interfaces Feature Guide for Routing Devices</i>

services (Application Identification)

Syntax	<code>services application-identification { ... }</code>
Hierarchy Level	[edit]
Release Information	services statement introduced before Junos OS Release 7.4. application-identification statement introduced in Junos OS Release 9.5.
Description	Define the services to be applied to traffic.
Options	application-identification —Use the values configured for application-identification properties.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Application Identification</i>

session-timeout (Application Identification)

Syntax	<code>session-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit services application-identification application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define session lifetime for the specified application in seconds.
Options	<i>seconds</i> —Duration of session. Default: 3600 Range: 1 through 604,800
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34

session-timeout (Interfaces)

Syntax	<code>session-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> services-options]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Define session lifetime globally for the Multiservices interface in seconds.
Options	<i>seconds</i> —Duration of session. Range: 4 through 86,400
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34

signature

Syntax	<pre>signature <i>name</i> { chain-order; maximum-transactions <i>number</i>; member <i>name</i> { context <i>value</i>; direction (any client-to-server server-to-client); pattern <i>dfa-pattern</i>; } order <i>number</i>; }</pre>
Hierarchy Level	[edit services application-identification nested-application name]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Identify the name of the custom nested application signature definition. The name must be unique with a maximum length of 32 characters.
Options	<i>name</i> —Name of the signature definition. The remaining statements are described separately.
Required Privilege Level	system—To view this statement in the configuration. system control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Application Identification for Nested Applications on page 41

signature-method-all-ports

Syntax	<pre>signature-method-all-ports</pre>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	<p>Run signature matching on all traffic in application-identification. This is called the signature-match mode.</p> <p>In the default mode, or fast-port-match mode, all traffic destined to well-known ports (up to 1024) immediately returns the final port match. However, the device runs signature matching for all traffic destined for port 80,</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Global APPID Properties on page 43

source

Syntax	<pre> source { ip address </prefix-length>; port-range { tcp [ports-and-port-ranges]; udp [ports-and-port-ranges]; } } </pre>
Hierarchy Level	[edit services application-identification rule <i>rule-name</i> address <i>address-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define source properties for application-identification rule processing.
Options	The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring APPID Rules on page 36

source-address (AACL)

Syntax	source-address <i>address</i> ;
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 9.5. IPv6 support introduced in Junos OS Release 12.2.
Description	Specify the source address for rule matching.
Options	address —Source IPv4 or IPv6 address or prefix value.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Configuring Match Conditions in AACL Rules on page 22

source-address-range

Syntax	source-address-range low <i>minimum-value</i> high <i>maximum-value</i> ;
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 9.5. IPv6 support introduced in Junos OS Release 12.2.
Description	Specify the source address range for rule matching.
Options	<i>minimum-value</i> —Lower boundary for the IPv4 or IPv6 address range. <i>maximum-value</i> —Upper boundary for the IPv4 or IPv6 address range.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Match Conditions in AACL Rules on page 22

source-prefix-list (Services AACL)

Syntax	source-prefix-list <i>list-name</i> ;
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i> from]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Specify the source prefix list for rule matching. You configure the prefix list by including the prefix-list statement at the [edit policy-options] hierarchy level.
Options	<i>list-name</i> —Source prefix list.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring Match Conditions in AACL Rules on page 22

source-prefix-list (Services IDS)

Syntax	<code>source-prefix-list <i>list-name</i> <except>;</code>
Hierarchy Level	<code>[edit services ids rule <i>rule-name</i> term <i>term-name</i> from]</code>
Release Information	Statement introduced in Junos OS Release 8.2.
Description	Specify the source prefix list for rule matching. You configure the prefix list by including the prefix-list statement at the [edit policy-options] hierarchy level.
Options	<p><i>list-name</i>—Destination prefix list.</p> <p>except—(Optional) Exclude the specified prefix list from rule matching.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Match Conditions in IDS Rules</i>• <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i>

term

Syntax `term term-name {`
 `from {`
 `application-group-any;`
 `application-groups [application-group-names];`
 `application-unknown;`
 `applications [application-names];`
 `destination-address address <any-unicast>;`
 `destination-address-range low minimum-value high maximum-value;`
 `destination-prefix-list list-name;`
 `nested-application-unknown;`
 `source-address address <any-unicast>;`
 `source-address-range low minimum-value high maximum-value;`
 `source-prefix-list list-name;`
 `}`
 `then {`
 `(accept | discard);`
 `count (application | application-group | application-group-any | nested-application |`
 `none);`
 `forwarding-class class-name;`
 `policer policer-name;`
 `}`
 `}`

Hierarchy Level `[edit services aacl rule rule-name]`

Release Information Statement introduced in Junos OS Release 9.5.

Description Define the AACL term properties.

Options *term-name*—Identifier for the term.

The remaining statements are explained separately.

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

Related Documentation • [Configuring AACL Rules on page 21](#)

then

Syntax	<pre> then { (accept discard); count (application application-group application-group-any nested-application none); forwarding-class <i>class-name</i>; log <i>event-type</i>; policer <i>policer-name</i>; } </pre>
Hierarchy Level	[edit services aacl rule <i>rule-name</i> term <i>term-name</i>]
Release Information	<p>Statement introduced in Junos OS Release 9.5.</p> <p>policer statement added in Junos OS Release 9.6.</p> <p>The nested-application option for the count statement introduced in Junos OS Release 11.1.</p>
Description	Define the AACL term actions. You can configure the router to accept or discard the targeted traffic. The action modifiers (count and forwarding-class) are optional.
Options	<p>You can configure one of the following actions:</p> <ul style="list-style-type: none"> • accept—Accept the packets and all subsequent packets in flows that match the rules. • discard—Discard the packet and all subsequent packets in flows that match the rules. <p>When you select accept as the action, you can optionally configure one or both of the following action modifiers. No action modifiers are allowed with the discard action.</p> <ul style="list-style-type: none"> • count (application application-group application-group-any nested-application none)—For all accepted packets that match the rules, record a packet count using AACL statistics practices. You can specify one of the following options; there is no default setting: <ul style="list-style-type: none"> • application—Count the application that matched in the from clause. • application-group—Count the application group that matched in the from clause. • application-group-any—Count all application groups that match from application-group-any under the any group name. • nested-application—Count all nested applications that matched in the from clause. • none—Same as not specifying count as an action. • forwarding-class <i>class-name</i>—Specify the packets' forwarding-class name. <p>policer <i>policer-name</i>—Apply rate-limiting properties to the traffic as configured at the [edit firewall policer <i>policer-name</i>] hierarchy level. This configuration allows bit-rate and burst-size attributes to be applied to the traffic that are not supported by AACL rules. When you include a policer, the only allowed action is discard. For more information on policers, see the <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i>.</p>

Required Privilege	interface—To view this statement in the configuration.
Level	interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring ACL Rules on page 21• <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i>

traceoptions (Application Identification)

Syntax	<pre> traceoptions { file <i>filename</i> <files <i>number</i>> <match <i>regex</i>> <size <i>size</i>> <world-readable no-world-readable>; flag <i>flag</i>; no-remote-trace; } </pre>
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	<p>Configure application identification tracing options.</p> <p>To specify more than one tracing operation, include multiple flag statements.</p>
Options	<p>file <i>filename</i>—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log.</p> <p>files <i>number</i>—(Optional) Maximum number of trace files. When a trace file named <i>trace-file</i> reaches its maximum size, it is renamed <i>trace-file.0</i>, then <i>trace-file.1</i>, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.</p> <p>Range: 2 through 1000 files</p> <p>Default: 2 files</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option.</p> <p>flag—Tracing operation to perform. all is the only valid completion.</p> <ul style="list-style-type: none"> all—Trace all events. <p>match <i>regex</i>—(Optional) Regular expression for lines to be logged.</p> <p>no-world-readable—(Optional) Disallow any user to read the log file.</p> <p>size <i>size</i>—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named <i>trace-file</i> reaches this size, it is renamed <i>trace-file.0</i>. When the <i>trace-file</i> again reaches its maximum size, <i>trace-file.0</i> is renamed <i>trace-file.1</i> and <i>trace-file</i> is renamed <i>trace-file.0</i>. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.</p> <p>Syntax: xk to specify KB, xm to specify MB, or xg to specify GB</p> <p>Range: 10240 through 1073741824 or the maximum file size supported on your system</p> <p>If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.</p> <p>world-readable—(Optional) Allow any user to read the log file.</p>

Required Privilege	interface—To view this statement in the configuration.
Level	interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Tracing APPID Operations on page 46

traceoptions (Services Local Policy Decision Function)

Syntax	<pre> traceoptions { file <i>filename</i> <files <i>number</i>> <size <i>size</i>>; flag <i>flag</i>; no-remote-trace; } </pre>
Hierarchy Level	[edit services local-policy-decision-function], [edit system services local-policy-decision-function]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Configure local policy decision function (L-PDF) tracing options.
Options	<p>file <i>filename</i>—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log.</p> <p>files <i>number</i>—(Optional) Maximum number of trace files. When a trace file named <i>trace-file</i> reaches its maximum size, it is renamed <i>trace-file.0</i>, then <i>trace-file.1</i>, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.</p> <p>Range: 2 through 1000 files</p> <p>Default: 2 files</p> <p>If you specify a maximum number of files, you also must specify a maximum file size with the size option.</p> <p><i>flag</i>—Tracing operation to perform. To specify more than one flag, include multiple flag statements.</p> <ul style="list-style-type: none"> • all—Everything • configuration—Configuration traces • database—Database traces • general—Miscellaneous traces • gres—Graceful Routing Engine switchover (GRES) traces • ptsp-statistics—PTSP statistics traces • rtsock—Routing socket traces • statistics—Statistics traces • subscriber—Subscriber traces <p>no-remote-trace—Disable remote tracing.</p> <p>size <i>size</i>—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named <i>trace-file</i> reaches this size, it is renamed <i>trace-file.0</i>. When the <i>trace-file</i> again reaches its maximum size, <i>trace-file.0</i> is renamed</p>

trace-file.1 and ***trace-file*** is renamed ***trace-file.0***. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

Syntax: ***xk*** to specify KB, ***xm*** to specify MB, or ***xg*** to specify GB

Range: 10240 through 1073741824 or the maximum file size supported on your system

If you specify a maximum file size, you also must specify a maximum number of trace files with the ***files*** option.

Required Privilege Level	routing and trace—To view this statement in the configuration. routing-control and trace-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Tracing L-PDF Operations on page 67

type

Syntax	<code>type type;</code>
Hierarchy Level	[edit services application-identification application <i>application-name</i>] [edit services application-identification nested-application <i>name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define type of application, such as HTTP or FTP.
Options	<i>type</i> —Application type. This is a mandatory value and has a maximum length of 32 characters.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application Identification on page 34• Application Identification for Nested Applications on page 41

type-of-service

Syntax	<code>type-of-service <i>service-type</i>;</code>
Hierarchy Level	[edit services application-identification application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define the type of service by service objective. There is no default value.
Options	<p>The following <i>service-type</i> options are available:</p> <ul style="list-style-type: none"> • maximize-reliability—Service designed for maximum reliability in packet transmission. • maximize-throughput—Service designed for maximum throughput. • minimize-delay—Service designed for minimum delay in packet transmission. • minimize-monetary-cost—Service designed for minimum monetary cost.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Defining an Application Identification on page 34

url

Syntax	<code>url <i>url</i>;</code>
Hierarchy Level	[edit services application-identification download]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define the URL for application package downloads.
Options	<i>url</i> —Download URL.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring Automatic Download of Application Package Updates on page 46

CHAPTER 6

Operational Commands

- clear services application-aware-access-list statistics
- clear services application-identification application-system-cache
- clear services application-identification counter
- clear services flows
- clear services flows ip-action
- clear services local-policy-decision-function statistics
- request services application-identification application
- request services application-identification download
- request services application-identification download status
- request services application-identification group
- request services application-identification install
- request services application-identification install status
- show services application-aware-access-list flows
- show services application-identification application-system-cache
- show services application-identification counter
- show services application-identification group
- show services application-aware-access-list statistics
- show services application-identification application
- show services application-identification version
- show services flows
- show services local-policy-decision-function flows
- show services local-policy-decision-function statistics
- show services sessions

[clear services application-aware-access-list statistics](#)

Syntax	clear services application-aware-access-list statistics
Release Information	Command introduced in Junos OS Release 9.5.
Description	Clear application aware access list (AACL) statistics.
Options	This command has no options.
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none">• show services application-aware-access-list statistics on page 162

clear services application-identification application-system-cache

Syntax	clear services application-identification application-system-cache
Release Information	Command introduced in Junos OS Release 9.5.
Description	Clear entries from application system cache.
Options	This command has no options.
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none">• show services application-identification application-system-cache on page 155

clear services application-identification counter

Syntax	clear services application-identification counter
Release Information	Command introduced in Junos OS Release 9.5.
Description	Clear application identification counters.
Options	This command has no options.
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none">• show services application-identification counter on page 157

clear services flows

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

Release Information Command introduced in Junos OS Release 9.5.
ip-action option introduced in Junos OS Release 10.0.
application-protocol option introduced in Junos OS Release 11.1.

Description Clear flow session table entries.

Options none—Clear all flows.

application-protocol—(Optional) Clear flows for one of the following application protocols:

- **bootp**—Bootstrap protocol
- **dce-rpc**—Distributed Computing Environment-Remote Procedure Call protocols
- **dce-rpc-portmap**—Distributed Computing Environment-Remote Procedure Call protocols portmap service
- **dns**—Domain Name System protocol
- **exec**—Exec
- **ftp**—File Transfer Protocol
- **h323**—H.323 standards
- **icmp**—Internet Control Message Protocol
- **iiop**—Internet Inter-ORB Protocol
- **login**—Login
- **netbios**—NetBIOS
- **netshow**—NetShow
- **pptp**—Point-to-Point Tunneling Protocol
- **realaudio**—RealAudio
- **rpc**—Remote Procedure Call protocol
- **rpc-portmap**—Remote Procedure Call protocol portmap service
- **rtsp**—Real-Time Streaming Protocol
- **shell**—Shell

- **sip**—Session Initiation Protocol
- **snmp**—Simple Network Management Protocol
- **sqlnet**—SQLNet
- **talk**—Talk Program
- **tftp**—Trivial File Transfer Protocol
- **traceroute**—Traceroute
- **winframe**—WinFrame

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

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

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

ip-action—(Optional) Clear **ip-action** entries generated by the router to log, drop, or block traffic based on previous matches. The IP action options and targets are configured at the **[edit security idp idp-policy policy-name rulebase-ips rule rule-name then]** hierarchy level.

protocol—(Optional) Clear 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
- **icmp6**—Internet Control Message Protocol version 6
- **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 Transmission Protocol
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

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

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

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

Required Privilege Level clear

Related Documentation • [show services flows on page 168](#)

List of Sample Output [clear services flows on page 141](#)
[clear services flows ip-action on page 141](#)

Output Fields [Table 3 on page 141](#) lists the output fields for the **clear services flows** command. Output fields are listed in the approximate order in which they appear.

Table 3: clear services flows Output Fields

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

Sample Output

clear services flows

```
user@host> clear services flows
Interface  Service set      Flows removed
ms-2/0/0   IDP               1
```

clear services flows ip-action

```
user@host> clear services flows ip-action
Interface  Service set      Flows removed
ms-4/0/0   idp-service      1
```

clear services flows ip-action

Syntax	clear services flows ip-action
Release Information	Command introduced in Junos OS Release 10.0.
Description	Clear ip-action entries generated by the router to log, drop, or block traffic based on previous matches. The IP action options and targets are configured at the [edit security idp idp-policy <i>policy-name</i> rulebase-ips rule <i>rule-name</i> then] hierarchy level.
Options	This command has no options.
Required Privilege Level	clear
Output Fields	When you issue this command, you are provided feedback on the status of your request.


Sample Output

```
user@host> clear services flows ip-action
Interface  Service set      Flows removed
ms-4/0/0   idp-service      1
```

clear services local-policy-decision-function statistics

Syntax	clear services local-policy-decision-function statistics
Release Information	Command introduced in Junos OS Release 9.5.
Description	Clear local policy decision function (L-PDF) statistics.
Options	This command has no options.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• show services local-policy-decision-function statistics on page 177

request services application-identification application

Syntax	<pre>request services application-identification application copy <i>predefined-application-name</i> request services application-identification application [disable enable] <i>predefined-application-name</i> <no-commit></pre>
Release Information	Command introduced in Junos OS Release 11.4.
Description	Copy, disable, or enable a predefined application signature.
Options	<p>copy—(Optional) Copy a predefined application signature from the database to the configuration and change the name (for example, my:FTP). The ID and order will be generated automatically. Do not name your custom application signature with the “junos” prefix; this prefix is reserved for predefined application signatures. You can copy the same predefined application signature only once; duplicate custom signatures are not allowed. The copy command does not initiate signature recompilation.</p> <hr/> <div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">  </div> <div> <p>NOTE: In configuration mode, if an uncommitted action is pending, the request services application-identification application copy command will fail. Uncompiled application signatures are shown as uncommitted in the show services application-identification application [summary detail] command.</p> </div> </div> <hr/> <p>disable—(Optional) Disable a predefined application signature, initiate signature recompilation, and commit all pending uncompiled signatures to the configuration. Include the <no-commit> keyword to defer signature recompilation.</p> <p>The following conditions apply:</p> <ul style="list-style-type: none"> • You cannot disable a predefined application signature that is referenced by an active security policy or custom application signature. First modify or deactivate the policy or custom application signature. • If you disable an application signature, for example, junos:HTTP, that has nested applications, the nested applications will not be recognized. <p>enable—(Optional) Enable a predefined application signature, initiate signature recompilation, and commit all pending uncompiled signatures to the configuration. Include the <no-commit> keyword to defer signature recompilation.</p> <p>no-commit—(Optional) Enables you to enter multiple enable and disable commands before initiating signature recompilation (which takes some time) and committing the configuration. Uncompiled application signatures are shown as uncommitted in the show services application-identification application [summary detail] command.</p>
Required Privilege Level	maintenance

Related Documentation

- [show services application-identification application on page 164](#)

Output Fields

When you enter this command, the system provides feedback on the status of your request.

Sample Output

[request services application-identification application copy](#)

```
user@host> request services application-identification application junos:FTP copy
application package 63 copied successfully.
```

[request services application-identification application disable](#)

```
user@host> request services application-identification application disable junos:163
Please wait while we are re-compiling signatures ...
Please wait while we are re-compiling signatures ...
Please wait while we are re-compiling signatures ...
Please wait while we are re-compiling signatures ...
Disable application junos:163 succeed.
```

[request services application-identification application disable no-commit](#)

```
user@host> request services application-identification application disable
junos:FACEBOOK-SOCIALRSS no-commit
Disable application junos:FACEBOOK-SOCIALRSS succeed. It is not committed yet.
```

request services application-identification download

Syntax	request services application-identification download <version>;
Release Information	Statement introduced in Junos OS Release 10.2. Statement modified in Junos OS Release 11.4.
Description	Manually download the application package for Junos OS application identification. The application package is extracted from the IDP signature database and contains signature definitions for known applications, such as: DNS, Facebook, FTP, Skype, and SNMP.
Options	version —(Optional) Download a specific version of the application package from the Juniper security website. If you do not enter a version, the most recent version will be downloaded.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request services application-identification download status on page 147• request services application-identification install on page 150
List of Sample Output	request services application-identification download on page 146
Output Fields	When you enter this command, you are provided feedback on the status of your request.

Sample Output

request services application-identification download

```
user@host> request services application-identifications download
Please use command "request services application-identification download status"
to check status
```

request services application-identification download status

Syntax	request services application-identification download status
Release Information	Statement introduced in Junos OS Release 10.2. Statement modified in Junos OS Release 11.4.
Description	Check the download status of the application signature package. The downloaded application package is saved under /var/db/appid/sec-download/.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request services application-identification download on page 146
List of Sample Output	request services application-identification download status on page 147
Output Fields	When you enter this command, the system provides feedback on the status of your request.

Sample Output

request services application-identification download status

```
user@host> request services application-identifications download status
Application package 1608 is downloaded successfully.
```

request services application-identification group

Syntax	<code>request services application-identification group [copy disable enable] predefined-application-group-name</code>
Release Information	Command introduced in Junos OS Release 11.4.
Description	Copy, disable, or enable a predefined application group.
Options	<p>copy—(Optional) Copy a predefined application signature group from the database to the configuration and change the name (for example, my:FTP). The ID and order will be generated automatically. Do not name your custom application signature group with the “junos” prefix; this prefix is reserved for predefined application signature groups. You can copy the same predefined application signature group only once; duplicate custom signature groups are not allowed.</p> <p>NOTE: In configuration mode, if an uncommitted action is pending, the <code>request services application-identification group copy</code> command will fail.</p> <p>disable—(Optional) Disable a predefined application signature group.</p> <p>NOTE: You cannot disable a predefined application signature group that is referenced by an active security policy or custom application signature group. First modify or deactivate the policy or custom application signature group.</p> <p>enable—(Optional) Enable a predefined application signature group.</p>
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none"> show services application-identification group on page 160
Output Fields	When you enter this command, the system provides feedback on the status of your request.

Sample Output

request services application-identification group

```
user@host> request services application-identification group disable
junos:infrastructure:networking
Disable application group junos:infrastructure:networking succeed.
```

request services application-identification group

```
user@host> request services application-identification group enable
junos:infrastructure:networking
Enable application group junos:infrastructure:networking succeed.
```

request services application-identification group

```
user@host> request services application-identification group copy junos:infrastructure:networking
Please wait while we are copying group ...
Copy application group junos:infrastructure:networking succeed.
```

request services application-identification install

Syntax	request services application-identification install
Release Information	Statement introduced in Junos OS Release 11.4.
Description	Install the downloaded predefined application signature package.



NOTE: You can also use the `request security idp security-package [download | install]` command to download and install the predefined application signature package along with the IDP security package.

Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request services application-identification install status on page 151• request services application-identification download on page 146
Output Fields	When you enter this command, the system provides feedback on the status of your request.

Sample Output

```
user@host> request services application-identification install
Please use command "request services application-identification install status"
to check status and use command "request services application-identification
proto-bundle-status" to check protocol bundle status
```

request services application-identification install status

Syntax	request services application-identification install status
Release Information	Statement introduced in Junos OS Release 11.4.
Description	Display the status of the install operation.
Required Privilege Level	maintenance
Related Documentation	<ul style="list-style-type: none">• request services application-identification install on page 150
Output Fields	When you enter this command, the system provides feedback on the status of your request.

Sample Output

```
user@host> request services application-identification install status
Install application package version (1776) succeed.
```

show services application-aware-access-list flows

Syntax	show services application-aware-access-list flows <interface <i>interface-name</i>> <subscriber <i>subscriber-name</i>>
Release Information	Command introduced in Junos OS Release 10.1. Offload status for flows using Juniper Forwarding Mechanism (JFM) added in Junos OS Release 12.1.
Description	Display application-aware-access-list (AACL) flows. Offloading using JFM is supported only on MX Series routers with Modular Port Concentrators (MPCs).
Options	interface <i>interface-name</i> —Displays AACL flows for the specified interface(s) only. The keyword, interface, must be appended to the command. subscriber <i>subscriber-name</i> —Displays AACL flows for the specified subscriber(s) only. The keyword, subscriber, must be appended to the command.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> Application Aware Services Interfaces Feature Guide for Routing Devices
List of Sample Output	show services application-aware-access-list flows interface on page 153 show services application-aware-access-list flows subscriber on page 153 show services application-aware-access-list flows subscriber (Offloading Using JFM) on page 154
Output Fields	Table 4 on page 152 lists the output fields for the show services application-aware-access-list flows command. Output fields are listed in the approximate order in which they appear.

Table 4: show services application-aware-access-list flows Output Fields

Field Name	Field Description	Level of Output
5-tuple	This field comprises five components of the given flow. The components are: <ul style="list-style-type: none"> • Src IP • Dest IP • Src Port • Dest Port • Protocol 	All levels
Application-ID	The identification number associated with the application.	All levels
Dir	The direction in terms of input or output. <ul style="list-style-type: none"> • Input (I) • Output (O) 	All levels

Table 4: show services application-aware-access-list flows Output Fields (*continued*)

Field Name	Field Description	Level of Output
Off	The status of offload to Packet Forwarding Engine. The various options are: <ul style="list-style-type: none"> • Not Offloaded (-) • Policer Offloaded, Flow Not Offloaded (P) • Policer Not Offloaded, Flow Offloaded (F) • Policer and Offloaded (P+F) 	All levels
Off	The status of offload to Packet Forwarding Engine using JFM. The various options are: <ul style="list-style-type: none"> • Not Offloaded (-) • Offload requested but not completed (R) • Offload requested and completed (O) 	All levels
Actions	The types of actions displayed are: <ul style="list-style-type: none"> • discard: (D) • accept : A • accept, count [T]: C-A or C-G or C-T • accept, fwd-class [C]: FC • accept, policer [P]: P • accept, count [T], fwd-class [C]: C-T+FC • accept, count [T], policer [P]: C-T+P • accept, fwd-class [C], policer [P]: FC+P • accept, count[T],fwd-class[C],policer[P]: C-T+FC+P 	All levels

Sample Output

show services application-aware-access-list flows interface

```

user@host>show services application-aware-access-list flows interface ge-1/0/5.0
Interface: ge-1/0/5.0
service-set: aac1-countApps
service-set interface: ms-0/0/0
Currently active flows: 2
High watermark flows: 2

5-tuple                                     Application-ID
Dir Off Action
-----
--- ---
      1.0.5.2:47072-> 10.10.254.116:80 ,6 junos:http [64]
I  - C-T
      10.10.254.116:80 ->      1.0.5.2:47072,6 junos:http [64]
O  - C-T

```

show services application-aware-access-list flows subscriber

```

user@host>show services application-aware-access-list flows subscriber user@example.com
Subscriber: user@example.com

Service-set: ss1

```

```
Service-set interface: ms-2/0/0
Currently active flows: 4
High watermark flows: 40
```

5-tuple	Application-ID	Dir	Off	Action
150.100.100.100:20109->160.200.200.200:80,17	junos:http [64]	I	-	C-T+FC+P
160.200.200.200:80->150.100.100.100:20109,17	junos:http [64]	O	-	C-T+FC+P
150.100.100.100:20108->160.100.100.100:80,17	junos:http [64]	I	P+F	C-T+FC+P
160.100.100.100:80->150.100.100.100:20108,17	junos:http [64]	O	P+F	C-T+FC+P

show services application-aware-access-list flows subscriber (Offloading Using JFM)

```
user@host>show services application-aware-access-list flows subscriber user@example.com
Subscriber: user@example.com
```

```
Service-set: ssl
Service-set interface: ms-2/0/0
Currently active flows: 4
High watermark flows: 40
```

5-tuple	Application-ID	Dir	Off	Action
150.100.100.100:20109->160.200.200.200:80,17	junos:http [64]			I
- C-T+FC+P				
160.200.200.200:80 ->150.100.100.100:20109,17	junos:http [64]			O
- C-T+FC+P				
150.100.100.100:20108->160.100.100.100:80,17	junos:http [64]			I
R C-T+FC+P				
160.100.100.100:80 ->150.100.100.100:20108,17	junos:http [64]			O
O C-T+FC+P				

show services application-identification application-system-cache

Syntax `show application-identification application-system-cache
<interface interface-name>`

Release Information Command introduced in Junos OS Release 9.5.
interface option added in Junos OS Release 10.1.

Description Display the database of cached values stored by the application identification (APPID) system.



NOTE: The `show services application-identification application-system-cache` command gives the information only when the application identifier (AI) is matched with the signature.

Options `interface interface-name`—Displays the services interfaces to query.

Required Privilege Level view

List of Sample Output [show application-identification application-system-cache on page 156](#)

Output Fields [Table 5 on page 155](#) lists the output fields for the `show services application-identification application-system-cache` command. Output fields are listed in the approximate order in which they appear.

Table 5: show application-identification application-system-cache Output Fields

Field Name	Field Description	Level of Output
IP address	IP address.	All levels
Port	Port number.	All levels
Protocol	Protocol name.	All levels
Application	Application number.	All levels
CPU	CPU number	All levels

Sample Output

show application-identification application-system-cache

```
user@host> show application-identification application-system-cache interface ms-1/0/0  
pic: 2/0
```

IP address	Port	Protocol	Application	CPU
10.1.1.2	81	TCP	63	18

show services application-identification counter

Syntax	show services application-identification counter <interface <i>interface-name</i>>
Release Information	Command introduced in Junos OS Release 9.5. interface option added in Junos OS Release 10.1.
Description	Display application identification (APPID) counter statistics.
Options	interface <i>interface-name</i> —Displays the services interfaces to query.
Required Privilege Level	view
List of Sample Output	show services application-identification counter on page 158 show services application-identification counter on page 158
Output Fields	Table 6 on page 157 lists the output fields for the show services application-identification counter command. Output fields are listed in the approximate order in which they appear.

Table 6: show services application-identification counter Output Fields

Field Name	Field Description
pic	PIC number.
Total sessions	Total number of sessions.
Total identified sessions	Total number of identified sessions.
Total unidentified sessions	Total number of unidentified sessions.
Total identified-by-address sessions	Number of sessions identified by address.
Total unidentified-by-address sessions	Number of sessions not identified by address.
Total identified-by-port sessions	Number of sessions identified by port.
Total unidentified-by-port sessions	Number of sessions not identified by port.
Total identified-by-icmp sessions	Number of sessions identified by ICMP.
Total unidentified-by-icmp sessions	Number of sessions not identified by ICMP.
Total identified-by-ip-protocol sessions	Number of sessions identified by IP protocol.
Total unidentified-by-ip-protocol sessions	Number of sessions not identified by IP protocol.
Total identified-by-signature sessions	Number of sessions identified by signature.

Table 6: show services application-identification counter Output Fields (*continued*)

Field Name	Field Description
Total unidentified-by-signature sessions	Number of sessions not identified by signature.
Total unspecified encrypted sessions	Number of encrypted sessions not specified by normal processes.
Total encrypted P2P sessions	Number of encrypted point-to-point sessions.
Total application system cache hits	Number of sessions found in the application system cache.
Total application system cache misses	Number of sessions not found in the application system cache.
Total identified-by-protocol sessions	Number of sessions identified by protocol.
Total unidentified-by-protocol sessions	Number of sessions not identified by protocol.

Sample Output

show services application-identification counter

```

user@host> show services application-identification counter interface ms-1/0/0
Counter Statistics:
  pic: 1/1
  Total sessions: 11
  Total identified sessions: 11
  Total un-identified sessions: 0
Address Method
  Total identified-by-address sessions: 0
  Total unidentified-by-address sessions: 11
Port Method
  Total identified-by-port sessions: 1
  Total unidentified-by-port sessions: 0
  Total identified-by-icmp sessions: 0
  Total unidentified-by-icmp sessions: 0
  Total identified-by-ip-protocol sessions: 0
  Total unidentified-by-ip-protocol sessions: 0
Signature Method
  Total identified-by-signature sessions: 11
  Total unidentified-by-signature sessions: 0
  Total unspecified encrypted sessions: 2
  Total encrypted P2P sessions: 2
  Total application system cache hits: 10
  Total application system cache misses: 1
Protocol Method
  Total identified-by-protocol sessions: 0
  Total unidentified-by-protocol sessions: 0

```

show services application-identification counter

```

user@host> show services application-identification counter interface ams0
Counter Statistics:
  pic: ams0
  Total sessions: 20
  Total identified sessions: 20
  Total un-identified sessions: 0

```

```
Protocol Method
  Total identified-by-protocol sessions: 0
  Total un-identified-by-protocol sessions: 0
Address Method
  Total identified-by-address sessions: 0
  Total un-identified-by-address sessions: 0
Port Method
  Total identified-by-port sessions: 0
  Total un-identified-by-port sessions: 0
  Total identified-by-icmp sessions: 0
  Total un-identified-by-icmp sessions: 0
  Total identified-by-ip-protocol sessions: 0
  Total un-identified-by-ip-protocol sessions: 0
Signature Method
  Total identified-by-signature sessions: 20
  Total identified-by-signature uni-directional sessions: 0
  Total un-identified-by-signature sessions: 0
  Total application system cache hits: 0
  Total application system cache misses: 0
```

show services application-identification group

Syntax	show services application-identification group [detail <i>application-group name</i>] summary]
Release Information	Command introduced in Junos OS Release 11.4.
Description	Display detailed or summary information about a specified application signature group or all application signature groups. Both custom and predefined application signature groups can be displayed.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • request services application-identification group on page 148
List of Sample Output	show services application-identification group summary on page 160 show services application-identification group detail on page 161
Output Fields	Table 7 on page 160 lists the output fields for the show services application-identification group command. Output fields are listed in the approximate order in which they appear.

Table 7: show services application-identification group Output Fields

Field Name	Field Description
Description	Description of the specified application in the detailed display.
Group ID or ID	The unique ID number of an application signature or application signature group. ID numbers 1 through 32,767 are automatically generated for predefined application signatures and application signature groups; these IDs do not change. ID numbers for custom application signatures and application signature groups use ID numbers 32,768 to 65,534.
Disabled	The status of the application signature group and whether the signature method is currently used to identify this application. The default is No.
Application Group(s)	The application signature groups present.
Applications	The application signatures associated with this application signature group.

Sample Output

show services application-identification group summary

```

user@host> show services application-identification group summary
Application Group(s): 24
Application Groups                                Disabled  ID
my:enterprise                                    No        32770
junos:enterprise:voip                            No        25
junos:peer-to-peer:voip                         No        24
junos:peer-to-peer:chat                         No        23
junos:peer-to-peer:file-sharing                 No        22
...

```


show services application-identification group detail

```
user@host> show services application-identification group detail junos:social-networking
Description: Detection for social networking sites such as Myspace, Facebook and
similar.
Group ID: 8
Disabled: no
Application-groups:
    junos:social-networking:facebook;
    junos:social-networking:myspace;

Applications:
    junos:4CHAN;
    junos:ADULTFRIENDFINDER;
    junos:BAD00;
    junos:BEBO;
    junos:BLOGGER-POST;
    junos:BLOGSPOT-POST;
```

show services application-aware-access-list statistics

Syntax	show services application-aware-access-list statistics <interface <i>interface-name</i>> <subscriber <i>subscriber-name</i>>
Release Information	Command introduced in Junos OS Release 9.5.
Description	Display application-aware-access-list (AACL) statistics.
Options	interface <i>interface-name</i> —(Optional) Displays AACL statistics for the specified interface only. subscriber <i>subscriber-name</i> —(Optional) Displays AACL statistics for the specified subscriber only.
Required Privilege Level	view
List of Sample Output	show services application-aware-access-list statistics interface on page 163 show services application-aware-access-list statistics subscriber on page 163
Output Fields	Table 8 on page 162 lists the output fields for the show services application-aware-access-list statistics command. Output fields are listed in the approximate order in which they appear.

Table 8: show services application-aware-access-list statistics Output Fields

Field Name	Field Description	Level of Output
Interface	Interface name.	Subscriber option
Subscriber	Subscriber identifier.	Interface option
Service-set-interface	Service set interface name.	All levels
Service set	Service set name.	All levels
Application group	Application group identifier.	All levels
Packets in	Number of ingress packets.	All levels
Bytes in	Number of ingress bytes.	All levels
Packets out	Number of egress packets.	All levels
Bytes out	Number of egress bytes.	All levels

Sample Output

show services application-aware-access-list statistics interface

```
user@host> show services application-aware-access-list statistics interface ge-0/0/0.100
Subscriber: user@example.com
```

```
service-set: IDP
service-set interface: ms-2/0/0
```

Application group	Application	Packets in	Bytes in
Packets out	Bytes out		
6	junos:ftp [63] 346	5	334

show services application-aware-access-list statistics subscriber

```
user@host> show services application-aware-access-list statistics subscriber user@example.com
Interface: ge-1/1/0.0
```

```
Service-set-interface: ms-1/3/0
Service set: aacl-svc-set
```

Application-aware-access-list statistics

Application group	Packets in	Bytes in	Packets out	Bytes
out				
P2P	16284	400	32025	200
FTP	8700	20000	5231000	100

show services application-identification application

Syntax	show services application-identification application [detail <i>application-name</i>] summary]
Release Information	Command introduced in Junos OS Release 11.4.
Description	Display detailed information about a specified application signature, all application signatures, or a summary of the existing application signatures and nested application signatures. Both custom and predefined application signatures and nested application signatures can be displayed.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • request services application-identification application on page 144
List of Sample Output	show services application-identification application summary on page 165 show services application-identification application detail on page 165
Output Fields	Table 9 on page 164 lists the output fields for the show services application-identification application command. Output fields are listed in the approximate order in which they appear.

Table 9: show services application-identification application Output Fields

Field Name	Field Description
Application(s)	The number of application signatures present.
Nested Application(s)	The number of nested application signatures present.
Application Name	Name of the predefined or custom application signature. Must be a unique name with a maximum length of 32 characters.
Application Group	Name of the application signature group associated with this application signature/nested application signature. Must be a unique name with a maximum length of 32 characters.
Disabled	The status of the application signature or nested application signature and whether the signature method is currently used to identify this application. The default is No.
ID	The unique ID number of an application signature or nested application signature. ID numbers 1 through 32,767 are automatically generated for predefined application signatures and nested application signatures; these IDs do not change. ID numbers for custom application signatures and nested application signatures use ID numbers 32,768 to 65,534.
Order	A unique number used to specify priority when multiple patterns are matched for the same session. The lowest order number takes the highest priority.

Table 9: show services application-identification application Output Fields (*continued*)

Field Name	Field Description
Application Tags	General information about this application type, for example, associated risk factors, technology, type of traffic, and so on. Support of application signature tags is dependent on the version of the loaded signature database. Please refer to the Juniper Networks security portal for further information.
Port Mapping: Default ports	The default port for this application type.
Signature: Port range	Default ranges: TCP/0 through 65,535; UDP/0 through 65,535 (optional).
Client-to-server: DFA Pattern	Pattern-matching scheme used for client-to-server traffic. Maximum length is 1023 (optional).
Client-to-server: Regex Pattern	A compatible regular expression used to match client-to-server traffic.
Server to-client: DFA Pattern	Pattern-matching scheme used for server-to-client traffic. Maximum length is 1023 (optional).
Server-to-client: Regex Pattern	A compatible regular expression used to match server-to-client traffic.
Minimum Data	The minimum number of bytes or packets to apply to the DFA pattern. Default is 10; range is 4 through 1024.

Sample Output

show services application-identification application summary

```

user@host> show services application-identification application summary
Application(s): 150
Nested Application(s): 600

Application      Disabled      ID      Order
...             ...          ...     ...
junos:FTP        No           63      59
junos:HTTP       Yes          64      122
...             ...          ...     ...
my:APPLICATION-A No           700     730
my:APPLICATION-B No           701     731
...             ...          ...     ...

```

show services application-identification application detail

```

user@host> show services application-identification detail junos:FTP
Description: This signature detects the File Transfer Protocol (FTP), which
provides facilities for transferring files to and from remote computer systems.
It usually runs on TCP port 21.
Application ID: 63
Disabled: No
Number of Parent Group(s): 1
Application Groups:
  junos:file-server
Application Tags:
  characteristic      : Supports File Transfer

```

```
characteristic      : Known Vulnerabilities
characteristic      : Capable of Tunneling
risk                : 3
category            : FILE-SERVER
Port Mapping:
  Default ports: TCP/21
Signature:
  Port range: TCP/0-24,26-65535
  Client-to-server
    DFA Pattern:
      \[(USER|STAT|PORT|CHMOD|ACCOUNT|BYE|ASCII|GLOB|HELP|AUTH|SYST|QUIT|STOR|PASV|QWD|PWD|MDTM|FEAT|OPTS)\](\s|\x0d
      0a\x|\x0a\x).*
    Regex Pattern: None
  Server-to-client
    DFA Pattern: (220|230|331|530)[\s\-.]*
    Regex Pattern: None
  Minimum data client-to-server: 8
  Minimum data server-to-client: 8
  Order: 71
```

show services application-identification version

Syntax	show services application-identification version
Release Information	Command introduced in Junos OS Release 10.2.
Description	Display the Junos OS application package version.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• request services application-identification download on page 146
List of Sample Output	show services application-identification version on page 167

Sample Output

show services application-identification version

The following output shows that the application package version is 1608.

```
user@host> show services application-identification version
Application package version: 1608
```

show services flows

Syntax `show services flows`
 `<all | brief | extensive | 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 in Junos OS Release 9.5.
 all option introduced in Junos OS Release 11.1.
 application-protocol option introduced in Junos OS Release 11.1.

Description Display flow session table entries.

Options **none**—Display standard information about all flows.

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

application-protocol—(Optional) Display information about one of the following application protocols:

- **bootp**—Bootstrap protocol
- **dce-rpc**—Distributed Computing Environment-Remote Procedure Call protocols
- **dce-rpc-portmap**—Distributed Computing Environment-Remote Procedure Call protocols portmap service
- **dns**—Domain Name System protocol
- **exec**—Exec
- **ftp**—File Transfer Protocol
- **h323**—H.323 standards
- **icmp**—Internet Control Message Protocol
- **iiop**—Internet Inter-ORB Protocol
- **login**—Login
- **netbios**—NetBIOS
- **netshow**—NetShow
- **pptp**—Point-to-Point Tunneling Protocol
- **realaudio**—RealAudio
- **rpc**—Remote Procedure Call protocol

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



NOTE: The flows for the DCE RPC ALG match the flows for the DCE RPC Portmap ALG. The flows for the RPC ALG match the flows for the RPC Portmap ALG.

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 **rspnumber**.

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
- **icmp6**—Internet Control Message Protocol version 6
- **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 Transmission 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 flows on page 139](#)

List of Sample Output [show services flows on page 171](#)
[show services flows all on page 171](#)
[show services flows brief on page 172](#)
[show services flows extensive on page 172](#)
[show services flows application-protocol on page 172](#)
[show services flows count on page 172](#)
[show services flows destination-port on page 172](#)
[show services flows destination-prefix on page 173](#)
[show services flows interface on page 173](#)
[show services flows protocol on page 173](#)
[show services flows service-set on page 173](#)
[show services flows source-port on page 173](#)
[show services flows source-prefix on page 173](#)

Output Fields [Table 10 on page 170](#) lists the output fields for the **show services flows** command. Output fields are listed in the approximate order in which they appear.

Table 10: show services flows Output Fields

Field Name	Field Description	Level of Output
Interface	Name of the interface.	All levels
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.	All levels
Flow Count	Number of flows in a session.	count only

Table 10: show services flows Output Fields (*continued*)

Field Name	Field Description	Level of Output
Flow or Flow Prot	Protocol used for this flow.	All levels
Source	Source prefix of the flow in the format <i>source-prefix:port</i> . For ICMP flows, port information is not displayed.	All levels
Dest	Destination prefix of the flow. For ICMP flows, port information is not displayed.	All levels
State	Status of the flow: <ul style="list-style-type: none"> • 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. 	All levels
Dir	Direction of the flow: input (I) or output (O).	All levels
Frm count	Number of frames in the flow.	All levels
Byte count	Number of bytes in the flow.	extensive
Flow role	Flow role.	extensive
Timeout	Timeout value.	extensive
Flow path	Flow path: symmetric or asymmetric.	extensive

Sample Output

show services flows

```

user@host> show services flows
Interface: ms-2/0/0, Service set: IDP
Flow                                     State  Dir      Frm count
TCP      10.2.2.2:33656 ->      10.1.1.2:80    Forward I         6
TCP      10.1.1.2:80 ->      10.2.2.2:33656 Forward O         5
ICMP     10.1.1.2 ->      10.2.2.2      Forward I        102
ICMP     10.2.2.2 ->      10.1.1.2      Forward O        102
ICMP     10.2.2.2 ->      10.1.1.2      Forward I         97
ICMP     10.1.1.2 ->      10.2.2.2      Forward O         97

```

show services flows all

```

user@host> show services flows all
Interface: ms-2/0/0, Service set: idp-1
Flow                                     State  Dir      Frm count
TCP      10.1.1.2:32769 ->      20.1.1.2:80    Forward I    353431
TCP      20.1.1.2:80 ->      10.1.1.2:32769 Forward O    353429
TCP      10.1.1.2:32771 ->      20.1.1.2:80    Forward I    353562
TCP      20.1.1.2:80 ->      10.1.1.2:32771 Forward O    353560

```

TCP	10.1.1.2:32770	->	20.1.1.2:80	Forward	I	353577
TCP	20.1.1.2:80	->	10.1.1.2:32770	Forward	O	353575
TCP	10.1.1.2:32768	->	20.1.1.2:80	Forward	I	353610
TCP	20.1.1.2:80	->	10.1.1.2:32768	Forward	O	353608
TCP	10.1.1.2:32777	->	20.1.1.2:80	Forward	I	353625
TCP	20.1.1.2:80	->	10.1.1.2:32777	Forward	O	353624
TCP	10.1.1.2:32776	->	20.1.1.2:80	Forward	I	353643
TCP	20.1.1.2:80	->	10.1.1.2:32776	Forward	O	353642
TCP	10.1.1.2:32775	->	20.1.1.2:80	Forward	I	353658
TCP	20.1.1.2:80	->	10.1.1.2:32775	Forward	O	353657
TCP	10.1.1.2:32774	->	20.1.1.2:80	Forward	I	353676
TCP	20.1.1.2:80	->	10.1.1.2:32774	Forward	O	353674
TCP	10.1.1.2:32773	->	20.1.1.2:80	Forward	I	353692
TCP	20.1.1.2:80	->	10.1.1.2:32773	Forward	O	353690
TCP	10.1.1.2:32772	->	20.1.1.2:80	Forward	I	353704
TCP	20.1.1.2:80	->	10.1.1.2:32772	Forward	O	353702

show services flows brief

The output for the **show services flows brief** command is identical to that for the **show services flows** command. For sample output, see [show services flows](#).

show services flows extensive

```
user@host> show services flows extensive
Interface: ms-2/0/0, Service set: IDP
Flow                                     State  Dir  Frm count
TCP      10.2.2.2:33656 -> 10.1.1.2:80  Forward  I      6
  Byte count: 346
  Flow role: Unknown, Timeout: 0, Flow path: Asymmetric
TCP      10.1.1.2:80 -> 10.2.2.2:33656 Forward  O      5
  Byte count: 334
  Flow role: Unknown, Timeout: 0, Flow path: Symmetric
ICMP      10.1.1.2 -> 10.2.2.2  Forward  I     144
  Byte count: 12096
  Flow role: Unknown, Timeout: 0, Flow path: Symmetric
ICMP      10.2.2.2 -> 10.1.1.2  Forward  O     144
  Byte count: 12096
  Flow role: Unknown, Timeout: 0, Flow path: Symmetric
```

show services flows application-protocol

```
user@host> show services flows application-protocol dce-rpc
Interface: ms-2/0/0, Service set: ss-1
Flow                                     State  Dir  Frm count
TCP      192.168.200.65:1260 -> 192.168.200.69:5315 Forward  I     14
TCP      192.168.200.69:5315 -> 16.16.16.16:1031 Forward  O     11
TCP      192.168.200.65:1251 -> 192.168.200.69:1026 Forward  I      7
TCP      192.168.200.69:1026 -> 16.16.16.16:1029 Forward  O      5
```

show services flows count

```
user@host> show services flows count
Interface  Service set  Flow count
ms-2/0/0   IDP          6
```

show services flows destination-port

```
user@host> show services flows destination-port 80
```

```

Interface: ms-2/0/0, Service set: IDP
Flow      State  Dir  Frm count
TCP       10.2.2.2:33656 -> 10.1.1.2:80 Forward I      6

```

show services flows destination-prefix

```

user@host> show services flows destination-prefix 10.1.1.2
Interface: ms-2/0/0, Service set: IDP
Flow      State  Dir  Frm count
TCP       10.2.2.2:33656 -> 10.1.1.2:80 Forward I      6
ICMP      10.2.2.2      -> 10.1.1.2 Forward 0     137
ICMP      10.2.2.2      -> 10.1.1.2 Forward I     132

```

show services flows interface

```

user@host> show services flows interface ms-2/0/0
Interface: ms-2/0/0, Service set: IDP
Flow      State  Dir  Frm count
TCP       10.2.2.2:33656 -> 10.1.1.2:80 Forward I      6
TCP       10.1.1.2:80 -> 10.2.2.2:33656 Forward 0      5
ICMP      10.1.1.2      -> 10.2.2.2 Forward I     162
ICMP      10.2.2.2      -> 10.1.1.2 Forward 0     162
ICMP      10.2.2.2      -> 10.1.1.2 Forward I     157
ICMP      10.1.1.2      -> 10.2.2.2 Forward 0     157

```

show services flows protocol

```

user@host> show services flows protocol icmp
Interface: ms-2/0/0, Service set: IDP
Flow      State  Dir  Frm count
ICMP      10.1.1.2      -> 10.2.2.2 Forward I     202
ICMP      10.2.2.2      -> 10.1.1.2 Forward 0     202
ICMP      10.2.2.2      -> 10.1.1.2 Forward I     197
ICMP      10.1.1.2      -> 10.2.2.2 Forward 0     197

```

show services flows service-set

```

user@host> show services flows service-set sample
Interface: ms-2/0/0, Service set: sample
Flow      State  Dir  Frm count
TCP       10.2.2.2:33656 -> 10.1.1.2:80 Forward I      6
TCP       10.1.1.2:80 -> 10.2.2.2:33656 Forward 0      5
ICMP      10.1.1.2      -> 10.2.2.2 Forward I     220
ICMP      10.2.2.2      -> 10.1.1.2 Forward 0     220
ICMP      10.2.2.2      -> 10.1.1.2 Forward I     215
ICMP      10.1.1.2      -> 10.2.2.2 Forward 0     215

```

show services flows source-port

```

user@host> show services flows source-port 0
Interface: ms-2/0/0, Service set: IDP
Flow      State  Dir  Frm count
TCP       10.2.2.2:33656 -> 10.1.1.2:80 Forward I      6
TCP       10.1.1.2:80 -> 10.2.2.2:33656 Forward 0      5
ICMP      10.1.1.2      -> 10.2.2.2 Forward I     235
ICMP      10.2.2.2      -> 10.1.1.2 Forward 0     235
ICMP      10.2.2.2      -> 10.1.1.2 Forward I     230
ICMP      10.1.1.2      -> 10.2.2.2 Forward 0     230

```

show services flows source-prefix

```

user@host> show services flows source-prefix 10.2.2.2

```

Interface: ms-2/0/0, Service set: IDP

Flow				State	Dir	Frm count
TCP	10.2.2.2:33656	->	10.1.1.2:80	Forward	I	6
TCP	10.1.1.2:80	->	10.2.2.2:33656	Forward	O	5
ICMP	10.1.1.2	->	10.2.2.2	Forward	I	235
ICMP	10.2.2.2	->	10.1.1.2	Forward	O	235
ICMP	10.2.2.2	->	10.1.1.2	Forward	I	230
ICMP	10.1.1.2	->	10.2.2.2	Forward	O	230

show services local-policy-decision-function flows

Syntax	show services local-policy-decision-function flows (interface <i>interface-name</i> subscriber <i>subscriber-name</i>)
Release Information	Command introduced in Junos OS Release 9.5.
Description	Display local policy decision function (L-PDF) flows.
Options	interface <i>interface-name</i> —Display L-PDF flows for the specified interfaces only. subscriber <i>subscriber-name</i> —Display L-PDF flows for the specified subscribers only.
Required Privilege Level	view
List of Sample Output	show services local-policy-decision-function flows subscriber on page 176 show services local-policy-decision-function flows interface on page 176
Output Fields	Table 11 on page 175 lists the output fields for the show services local-policy-decision-function flows command. Output fields are listed in the approximate order in which they appear.

Table 11: show services local-policy-decision-function flows Output Fields

Field Name	Field Description
Interface	Interface name.
service-set	Service set name.
service-set-interface	Service set interface name.
Currently active flows	Number of currently active flows.
High watermark flows	Maximum number of flows.
Protocol	(With interface option) Protocol identifier.
Source address	(With interface option) Source address.
Source port	(With interface option) Source port.
Destination address	(With interface option) Destination address.
Destination port	(With interface option) Destination port.
Application	(With interface option) Application name.
Application group	(With interface option) Application group identifier.

Sample Output

show services local-policy-decision-function flows subscriber

```
user@host> show services local-policy-decision-function flows subscriber user@example.com
Interface: ge-0/0/5.26

service-set: aac1_ms30
service-set interface: ms-3/0/0

Currently active flows: 0
High watermark flows: 0
```

show services local-policy-decision-function flows interface

```
user@host> show services local-policy-decision-function flows interface ge-1/1/0
Interface: ge-1/1/0.0

service-set: IDP
service-set interface: ms-2/0/0

Currently active flows: 2
High watermark flows: 2
```

Protocol	Source address	Source port	Destination address	Destination port
Application		Application group		
tcp	10.1.1.2	81	20.1.1.2	32813
	junos:ftp [63]	unknown [1023]		
tcp	20.1.1.2	32813	10.1.1.2	81
	junos:ftp [63]	unknown [1023]		

show services local-policy-decision-function statistics

Syntax	<code>show services local-policy-decision-function statistics (interface <i>interface-name</i> subscriber <i>subscriber-name</i>)</code>
Release Information	Command introduced in Junos OS Release 9.5.
Description	Display local-policy-decision-function (L-PDF) statistics.
Options	<p><code>interface <i>interface-name</i></code>—Display L-PDF statistics for the specified interface only.</p> <p><code>subscriber <i>subscriber-name</i></code>—Display L-PDF statistics for the specified subscriber only.</p>
Required Privilege Level	view
List of Sample Output	<p>show services local-policy-decision-function statistics interface on page 177</p> <p>show services local-policy-decision-function statistics subscriber on page 178</p>
Output Fields	<p>Table 12 on page 177 lists the output fields for the <code>show services local-policy-decision-function statistics</code> command. Output fields are listed in the approximate order in which they appear.</p>

Table 12: show services local-policy-decision-function statistics Output Fields

Field Name	Field Description
Interface	Interface name.
service-set	Service set name.
service-set-interface	Service set interface name.
Application group	Application group identifier.
Application	Application name.
Packets in	Number of ingress packets.
Bytes in	Number of ingress bytes.
Packets out	Number of egress packets.
Bytes out	Number of egress bytes.

Sample Output

show services local-policy-decision-function statistics interface

```
user@host> show services local-policy-decision-function statistics interface ge-1/1/0
```

Interface: ge-1/1/0.0

service-set: IDP

service-set interface: ms-2/0/0

Application group	Application	Packets in	Bytes in
Packets out	Bytes out		
	junos:ftp [63]	5	334
6	346		

show services local-policy-decision-function statistics subscriber

user@host> show services local-policy-decision-function statistics subscriber user@example.com

Service-set-interface: ms-1/3/0

Service set: aac1-svc-set

Application-aware-access-list statistics

Application group	Packets in	Bytes in	Packets out	Bytes
out				
P2P		400	32025	200
	16284			
FTP		20000	5231000	100
	8700			

show services sessions

Syntax show services sessions
 <brief | extensive | 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*>
 <utilization>

Release Information Command introduced in Junos OS Release 10.4.

Description Display session information.



NOTE: On MX Series routers (with interchassis redundancy configured), the idle timeout for every flow is displayed in the **show services session extensive** and **show services flows extensive** commands.

Options **none**—Display standard information about all sessions.

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

application-protocol—(Optional) Display information about one of the following application protocols:

- **bootp**—Bootstrap protocols
- **dce-rpc**—Distributed Computing Environment-Remote Procedure Call protocols
- **dce-rpc-portmap**—Distributed Computing Environment-Remote Procedure Call protocols portmap service
- **dns**—Domain Name System protocol
- **exec**—Remote Execution Protocol
- **ftp**—File Transfer Protocol
- **h323**—H.323
- **icmp**—ICMP
- **icmpv6**—ICMPv6
- **iiop**—Internet Inter-ORB Protocol
- **ip**—IP

- **login**—LOGIN
- **netbios**—NETBIOS
- **netshow**—NETSHOW
- **pptp**—Point-to-Point Tunneling Protocol
- **realaudio**—RealAudio
- **rpc**—Remote Procedure Call protocol
- **rpc-portmap**—Remote Procedure Call protocol portmap service
- **rtsp**—Real-Time Streaming Protocol
- **rsh**—Remote Shell
- **sip**—Session Initiation Protocol
- **shell**—Shell
- **snmp**—SNMP
- **sql**—SQLNet
- **talk**—Talk Program
- **tftp**—Trivial File Transfer Protocol
- **traceroute**—Traceroute
- **winframe**—WinFrame



NOTE: You can use the **none** option with the **show services sessions count application-protocol** command to display information about sessions other than ALG sessions.

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 **rspnumber**. On J Series routers, *interface-name* is **ms-pim/O/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
- **icmp6**—Internet Control Message Protocol version 6
- **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 Transmission 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.

utilization—(Optional) Display statistical details about session utilization.

Required Privilege Level

view

List of Sample Output

[show services sessions on page 183](#)
[show services sessions brief on page 183](#)
[show services sessions extensive on page 183](#)
[show services sessions terse on page 183](#)
[show services sessions application-protocol on page 183](#)
[show services sessions count on page 185](#)
[show services sessions destination-port on page 185](#)
[show services sessions destination-prefix on page 185](#)
[show services sessions interface on page 185](#)
[show services sessions protocol on page 186](#)
[show services sessions service-set on page 186](#)
[show services sessions source-port on page 186](#)
[show services sessions source-prefix on page 186](#)

Output Fields

Table 13 on page 182 lists the output fields for the **show services sessions** command. Output fields are listed in the approximate order in which they appear.

Table 13: show services sessions Output Fields

Field Name	Field Description
Interface	Name of the interface.
Session ID	Session ID that uniquely identifies the session.
ALG	Name of the application.
Flags	Session flag for the ALG: <ul style="list-style-type: none"> • 0x1—Found an existing session. • 0x2—Reached session or flow limit. • 0x3—No memory available for new sessions. • 0x4—No free session ID available. • 0x0000—No session ID found.
IP Action	Flag indicating whether IP action has been set for the session..
Offload	Flag indicating whether the session has been offloaded to the Packet Forwarding Engine.
Asymmetric	Flag indicating whether the session is uni-directional.
Service set	Name of a service set. Individual empty service sets are not displayed.
Sessions Count	Number of sessions.
Flow or Flow Prot	Protocol used for this session.
Source	Source prefix of the flow in the format <i>source-prefix:port</i> . 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: <ul style="list-style-type: none"> • 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. • Bypass—Bypass packets in the flow. • Unknown—Unknown flow status.
Packet Direction	Direction of the flow: ingress (I), egress (O) or unknown.
Frm count	Number of frames in the flow.

Sample Output

show services sessions

```

user@host> show services sessions
ms-2/0/0
Session: 293, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    10.10.10.2:43677 ->    10.20.20.1:53    Forward I        1
UDP    10.20.20.1:53    ->    1.1.1.1:43677 Forward 0        1
Session: 53, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    10.10.10.2:37494 ->    10.20.20.1:53    Forward I        1
UDP    10.20.20.1:53    ->    10.11.11.11:37494 Forward 0        1
Session: 66, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    10.10.10.2:48161 ->    10.20.20.1:53    Forward I        1
UDP    10.20.20.1:53    ->    10.11.11.11:48161 Forward 0        1
Session: 17, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    10.10.10.2:38908 ->    10.20.20.1:53    Forward I        1
UDP    10.20.20.1:53    ->    10.11.11.11:38908 Forward 0        1
Session: 42, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    10.10.10.2:58189 ->    10.20.20.1:53    Forward I        1
UDP    10.20.20.1:53    ->    10.11.11.11:58189 Forward 0        1

```

show services sessions brief

The output for the **show services flows brief** command is identical to that for the **show services sessions** command. For sample output, see [show services sessions on page 183](#).

show services sessions extensive

```

user@host> show services sessions extensive
ms-0/1/0
Session: 2, ALG: 0, Flags: 0x0080, IP Action: no, Offload: no
NAT Plugin Data:
  NAT Action: Translation Type - DYNAMIC NAT44
  NAT source   3.1.1.2    ->    10.10.10.127
TCP    3.1.1.2:52145 ->    4.1.1.2:23    Forward I        22
  Byte count: 1483
  Flow role: Unknown, Timeout: 0
TCP    4.1.1.2:23    ->    10.10.10.127:52145 Forward 0        18
  Byte count: 2712
  Flow role: Unknown, Timeout: 0

```

show services sessions terse

```

user@host> show services sessions terse
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP    10.2.2.2:52138 ->    10.1.1.2:21    Forward I        33
TCP    10.1.1.2:21    ->    10.2.2.2:52138 Forward 0        31

```

show services sessions application-protocol

This command has the same output for the **rpc**, **dce-rpc**, **rpc-portmap** and **dce-rpc-portmap** ALGs.

```

user@host> show services sessions application-protocol dce-rpc
Interface name: ms-1/1/0
Session: 8, ALG: portmapper, Flags: 0x1800, IP Action: no, Offload: no
UDP    192.168.203.198:1019 ->192.168.203.194:2049 Forward I        4

```

```

UDP    192.168.203.194:2049 ->192.168.203.198:1019 Forward 0      4
Session: 7, ALG: portmapper, Flags: 0x1800, IP Action: no, Offload: no
UDP    192.168.203.198:954 ->192.168.203.194:613 Forward I      1
UDP    192.168.203.194:613 ->192.168.203.198:954 Forward 0      1
Session: 6, ALG: portmapper, Flags: 0x1800, IP Action: no, Offload: no
UDP    192.168.203.198:53836 ->192.168.203.194:613 Forward I      1
UDP    192.168.203.194:613 ->192.168.203.198:53836 Forward 0      1
Session: 5, ALG: portmapper, Flags: 0x1000, IP Action: no, Offload: no
UDP    192.168.203.198:59813 ->192.168.203.194:111 Forward I      1
UDP    192.168.203.194:111 ->192.168.203.198:59813 Forward 0      1
Session: 4, ALG: portmapper, Flags: 0x1800, IP Action: no, Offload: no
UDP    192.168.203.198:36595 ->192.168.203.194:2049 Forward I      1
UDP    192.168.203.194:2049 ->192.168.203.198:36595 Forward 0      1
Session: 3, ALG: portmapper, Flags: 0x1000, IP Action: no, Offload: no
UDP    192.168.203.198:56050 ->192.168.203.194:111 Forward I      1
UDP    192.168.203.194:111 ->192.168.203.198:56050 Forward 0      1

```

user@host> show services sessions application-protocol dns

Interface name: ms-2/0/0

```

Session: 293, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    50.50.50.2:43677 -> 60.60.60.10:53 Forward I      1
UDP    60.60.60.10:53 -> 1.1.1.1:43677 Forward 0      1
Session: 53, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    50.50.50.2:37494 -> 60.60.60.10:53 Forward I      1
UDP    60.60.60.10:53 -> 1.1.1.1:37494 Forward 0      1
Session: 66, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    50.50.50.2:48161 -> 60.60.60.10:53 Forward I      1
UDP    60.60.60.10:53 -> 1.1.1.1:48161 Forward 0      1
Session: 17, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    50.50.50.2:38908 -> 60.60.60.10:53 Forward I      1
UDP    60.60.60.10:53 -> 1.1.1.1:38908 Forward 0      1
Session: 42, ALG: 16, Flags: 0x0040, IP Action: no, Offload: no
UDP    50.50.50.2:58189 -> 60.60.60.10:53 Forward I      1
UDP    60.60.60.10:53 -> 1.1.1.1:58189 Forward 0      1

```

user@host> show services sessions application-protocol ftp

Interface name: ms-4/1/0

```

Session: 1, ALG: 1, Flags: 0x0040, IP Action: no, Offload: no
TCP    30.1.1.1:32843 -> 20.1.1.1:21 Forward I      26
TCP    20.1.1.1:21 -> 1.1.1.0:32843 Forward 0      30

```

user@host> show services sessions application-protocol pptp

Interface name: ms-2/0/0

```

Session: 3, ALG: pptp, Flags: 0x2800, IP Action: no, Offload: no, Asymmetric: no
GRE    40.40.40.10:0 -> 15.15.15.10:0 Forward 0      21
GRE    15.15.15.10:0 -> 40.40.40.10:65000 Forward I      0
Session: 2, ALG: pptp, Flags: 0x2800, IP Action: no, Offload: no, Asymmetric: no
GRE    15.15.15.10:0 -> 40.40.40.10:49913 Forward I      88
GRE    40.40.40.10:49913 -> 15.15.15.10:65001 Forward 0      0
Session: 1, ALG: pptp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP    15.15.15.10:1511 -> 40.40.40.10:1723 Forward I      13
TCP    40.40.40.10:1723 -> 15.15.15.10:1511 Forward 0      12

```

user@host> show services sessions application-protocol rtsp

Interface name: ms-0/1/0

```

Session: 13, ALG: rtsp, Flags: 0x0800, IP Action: no, Offload: no
UDP    9.1.0.2:5004 -> 9.0.0.2:3989 Forward 0      152
UDP    9.0.0.2:3989 -> 3.1.2.1:5004 Forward I      0
Session: 9, ALG: rtsp, Flags: 0x0800, IP Action: no, Offload: no
UDP    9.1.0.2:5004 -> 9.0.0.2:3986 Forward 0      3
UDP    9.0.0.2:3986 -> 3.1.2.1:5004 Forward I      0

```

user@host> show services sessions application-protocol rsh


```

Interface name: ms-2/0/0
Session: 3, ALG: 2, Flags: 0x0840, IP Action: no, Offload: no
TCP    60.60.60.10:1023 ->    50.50.50.2:1020 Forward 0      4
TCP    50.50.50.2:1020 ->    60.60.60.10:1023 Forward I      3
Session: 1, ALG: 2, Flags: 0x0040, IP Action: no, Offload: no
TCP    50.50.50.2:1021 ->    60.60.60.10:514 Forward I    1331
TCP    60.60.60.10:514 ->    50.50.50.2:1021 Forward 0    2485

user@host> show services sessions application-protocol sip
Interface name: ms-2/0/0
Session: 4, ALG: sip, Flags: 0x0800, IP Action: no, Offload: no
UDP    20.1.1.2:6000 ->    30.1.1.2:12682 Forward I      246
UDP    30.1.1.2:12682 ->    70.1.1.2:6000 Forward 0      0
Session: 1, ALG: sip, Flags: 0x0000, IP Action: no, Offload: no
UDP    20.1.1.2:5060 ->    30.1.1.2:5060 Forward I      10
UDP    30.1.1.2:5060 ->    70.1.1.2:5060 Forward 0      9

user@host> show services sessions application-protocol sql
Interface name: ms-2/0/0
Session: 3934, ALG: sqlnet, Flags: 0x0800, IP Action: no, Offload: no
TCP    50.50.50.2:39754 ->    40.40.40.10:1408 Forward I      26
TCP    40.40.40.10:1408 ->    1.1.1.1:39754 Forward 0      23

user@host> show services sessions application-protocol talk
Interface name: ms-0/2/0
Session: 4, ALG: 65, Flags: 0x0800, IP Action: no, Offload: no
TCP    2.2.2.2:36888 ->    1.1.1.2:33294 Forward 0      4
TCP    1.1.1.2:33294 ->    2.2.2.2:36888 Forward I      3
Session: 7, ALG: 65, Flags: 0x0800, IP Action: no, Offload: no
UDP    2.2.2.2:1165 ->    1.1.1.2:518 Forward 0      1
UDP    1.1.1.2:518 ->    2.2.2.2:1165 Forward I      1
Session: 8, ALG: 65, Flags: 0x0000, IP Action: no, Offload: no
UDP    1.1.1.2:1509 ->    2.2.2.2:518 Forward I      3
UDP    2.2.2.2:518 ->    1.1.1.2:1509 Forward 0      3
Session: 6, ALG: 0, Flags: 0x0000, IP Action: no, Offload: no
UDP    1.1.1.1:123 ->    1.1.1.2:123 Forward 0      4

```

show services sessions count

```

user@host> show services sessions count
Interface  Service set      Sessions count
ms-1/1/0   ss                2

```

show services sessions destination-port

```

user@host> show services sessions destination-port 21
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP    10.2.2.2:52138 ->    10.1.1.2:21 Forward I      25
TCP    10.1.1.2:21 ->    10.2.2.2:52138 Forward 0      24

```

show services sessions destination-prefix

```

user@host> show services sessions destination-prefix 10.1.1.2
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP    10.2.2.2:52138 ->    10.1.1.2:21 Forward I      25
TCP    10.1.1.2:21 ->    10.2.2.2:52138 Forward 0      24

```

show services sessions interface

```

user@host> show services sessions interface ms-1/1/0

```

```
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP      10.2.2.2:52138 ->      10.1.1.2:21    Forward I      30
TCP      10.1.1.2:21    ->      10.2.2.2:52138 Forward 0      29
```

show services sessions protocol

```
user@host> show services sessions protocol tcp
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP      10.2.2.2:52138 ->      10.1.1.2:21    Forward I      30
TCP      10.1.1.2:21    ->      10.2.2.2:52138 Forward 0      29
```

show services sessions service-set

```
user@host> show services sessions service-set sample
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP      10.2.2.2:52138 ->      10.1.1.2:21    Forward I      33
TCP      10.1.1.2:21    ->      10.2.2.2:52138 Forward 0      31
```

show services sessions source-port

```
user@host> show services sessions source-port 21
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP      10.2.2.2:52138 ->      10.1.1.2:21    Forward I      33
TCP      10.1.1.2:21    ->      10.2.2.2:52138 Forward 0      31
```

show services sessions source-prefix

```
user@host> show services sessions source-prefix 10.2.2.2
ms-1/1/0
Session: 1, ALG: ftp, Flags: 0x2000, IP Action: no, Offload: no, Asymmetric: no
TCP      10.2.2.2:52138 ->      10.1.1.2:21    Forward I      33
TCP      10.1.1.2:21    ->      10.2.2.2:52138 Forward 0      31
```

CHAPTER 7

Index

- [Index on page 189](#)

Index

Symbols

#, comments in configuration statements.....	xiv
(), in syntax descriptions.....	xiv
< >, in syntax descriptions.....	xiv
[], in configuration statements.....	xiv
{ }, in configuration statements.....	xiv
(pipe), in syntax descriptions.....	xiv

A

AACL	
action statements.....	24
applications.....	22
best-effort application	
identification.....	18, 31, 53, 59
example configuration.....	26
logging flows.....	25
match conditions.....	22
rules.....	26
statistics	
clearing.....	136
aacl-fields statement.....	75
aacl-statistics-profile statement.....	76
address statement	
APPID	
usage guidelines.....	36
application rule.....	77
APPID	
best-effort application	
identification.....	18, 31, 53, 59
example configuration.....	48
application statement.....	78, 79
APPID	
usage guidelines.....	34
application-aware-access-list See aacl	
application-aware-access-list-fields	
statement.....	80
application-group statement.....	81
APPID	
usage guidelines.....	40

application-group-any statement.....	81
AACL	
usage guidelines.....	22
application-groups statement.....	82, 103
AACL	
usage guidelines.....	22
APPID	
usage guidelines.....	40
application-system-cache-timeout statement.....	83
APPID	
usage guidelines.....	43
applications statement	
AACL.....	83
usage guidelines.....	22
APPID	
usage guidelines.....	40
application identification.....	84
asymmetrical routing support	
APPID.....	45
automatic statement.....	84
APPID	
usage guidelines.....	46

B

best-effort application identification.....	18, 31, 53, 59
braces, in configuration statements.....	xiv
brackets	
angle, in syntax descriptions.....	xiv
square, in configuration statements.....	xiv
bypass-traffic-on-exceeding-flow-limits	
statement.....	85

C

chain-order statement	
nested applications.....	85
clear services application-aware-access-list	
statistics command.....	136
clear services application-identification	
application-system-cache command.....	137
clear services application-identification counter	
command.....	138
clear services flows command.....	139
clear services flows ip-action command.....	142
clear services local-policy-decision-function	
statistics command.....	143
command-name command.....	152
comments, in configuration statements.....	xiv
context statement	
nested applications.....	86

conventions	
text and syntax.....	xiii
curly braces, in configuration statements.....	xiv
customer support.....	xv
contacting JTAC.....	xv

D

data session identification	
APPID.....	37
destination statement	
APPID	
usage guidelines.....	36
application identification rule.....	86
destination-address statement	
AACL.....	87
usage guidelines.....	22
destination-address-range statement	
AACL.....	87
usage guidelines.....	22
destination-prefix-list statement	
AACL.....	88
usage guidelines.....	22
direction statement	
nested applications.....	88
disable statement	
APPID	
usage guidelines.....	34, 40
application.....	89
application group.....	89
port mapping.....	89
disable-global-timeout-override statement.....	90
usage guidelines.....	34
documentation	
comments on.....	xv
download statement	
APPID.....	90
usage guidelines.....	46

E

enable-asymmetric-traffic-processing	
statement.....	91
enable-heuristics statement.....	91
usage guidelines.....	44
encrypted traffic identification	
APPID.....	44
event policy	
all (tracing flag)	
APPID.....	48

F

file statement	
L-PDF statistics.....	92
flows	
access-list.....	152
list-flows.....	152
font conventions.....	xiii
from statement	
AACL.....	93
usage guidelines.....	21

H

heuristics support	
APPID.....	44

I

idle-timeout statement.....	94
APPID	
usage guidelines.....	34
ignore-errors statement.....	94
usage guidelines.....	37
inactivity-non-tcp-timeout statement.....	96
usage guidelines.....	34
inactivity-tcp-timeout statement.....	96
usage guidelines.....	34
index statement.....	95
APPID	
usage guidelines.....	34, 40
nested applications.....	95
ip statement	
APPID	
usage guidelines.....	36
application identification.....	97
ip-action	
clearing.....	142

L

L-PDF	
best-effort application	
identification.....	18, 31, 53, 59
statistics	
clearing.....	143
list-flows.....	152
See also list-statistics	
local-policy-decision-function statement.....	98
log output	
APPID.....	47

M

manuals	
comments on.....	xv
match-direction statement	
AACL.....	99
usage guidelines.....	22
max-checked-bytes statement.....	100
APPID	
usage guidelines.....	43
maximum-transactions statement	
nested applications.....	100
member statement	
nested applications.....	101
min-checked-bytes statement.....	101
APPID	
usage guidelines.....	43

N

nested-application statement	
APPID.....	102
usage guidelines.....	41
nested-application-settings statement	
APPID.....	103
no-application-identification statement.....	104
APPID	
usage guidelines.....	43
no-application-system-cache statement.....	105
APPID	
usage guidelines.....	43
no-clear-application-system-cache statement.....	105
APPID	
usage guidelines.....	43
no-nested-application statement.....	106
usage guidelines.....	42
no-protocol-method statement.....	106
APPID	
usage guidelines.....	43
no-signature-based statement.....	107
APPID	
usage guidelines.....	43

O

order statement.....	107
APPID	
usage guidelines.....	36

P

parentheses, in syntax descriptions.....	xiv
--	-----

pattern statement	
nested applications.....	108
policy-decision-statistics-profile statement.....	109
port-mapping statement.....	110
port-range statement.....	110
APPID	
usage guidelines.....	36
profile statement	
APPID	
usage guidelines.....	39
application identification.....	111
protocol statement	
nested applications.....	111

R

request services application-identification	
application command.....	144
request services application-identification	
download command.....	146
request services application-identification	
download status command.....	147
request services application-identification group	
command.....	148
request services application-identification install	
command.....	150
request services application-identification install	
status command.....	151
rule statement	
AACL.....	112
usage guidelines.....	21
APPID	
usage guidelines.....	36
rule-set statement	
AACL.....	114
usage guidelines.....	26
APPID	
usage guidelines.....	36
application identification.....	115

S

senable-asymmetric-traffic-processing statement	
usage guidelines.....	45
service-set statement.....	118
services statement	
AACL	
usage guidelines.....	71
APPID	
usage guidelines.....	72

session-timeout statement.....	121	support-uni-directional-traffic statement.....	117
usage guidelines.....	34	usage guidelines.....	45
show application-identification		syntax conventions.....	xiii
application-system-cache command.....	155		
show services application-aware-access-list			
statistics command.....	162		
show services application-identification application			
command.....	164		
show services application-identification counter			
command.....	157		
show services application-identification group			
command.....	160		
show services application-identification			
version.....	167		
show services flows command.....	168		
show services local-policy-decision-function flows			
command.....	175		
show services local-policy-decision-function			
statistics command.....	177		
show services sessions command.....	179		
signature statement			
nested applications.....	122		
signature-method-all-ports statement.....	122		
AAPID			
usage guidelines.....	43		
source statement			
AAPID			
usage guidelines.....	36		
application identification rule.....	123		
source-address statement			
AACL.....	123		
usage guidelines.....	22		
source-address-range statement			
AACL.....	124		
usage guidelines.....	22		
source-prefix-list statement			
AACL.....	124		
usage guidelines.....	22		
IDS.....	125		
stateful firewall use with APPID.....	37		
statistics			
AACL			
clearing.....	136		
L-PDF			
clearing.....	143		
statistics statement			
L-PDF.....	117		
support, technical See technical support			

T

technical support	
contacting JTAC.....	xv
term statement	
AACL.....	126
usage guidelines.....	21
then statement	
AACL.....	127
usage guidelines.....	21
traceoptions statement	
application identification.....	129
L-PDF.....	131
tracing flags	
event policy	
all.....	48
tracing operations	
APPID.....	46
type statement.....	132
APPID	
usage guidelines.....	34
type-of-service statement.....	133
APPID	
usage guidelines.....	34

U

undirectional traffic support	
APPID.....	45
url statement.....	133
APPID	
usage guidelines.....	46