



Application Identification



Published: 2013-08-29

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

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

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.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Application Identification

Copyright © 2013, 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	ix
	Documentation and Release Notes	ix
	Supported Platforms	ix
	Using the Examples in This Manual	ix
	Merging a Full Example	x
	Merging a Snippet	x
	Documentation Conventions	xi
	Documentation Feedback	xiii
	Requesting Technical Support	xiii
	Self-Help Online Tools and Resources	xiii
	Opening a Case with JTAC	xiv
Part 1	Overview	
Chapter 1	Application Identification	3
	APPID Overview	3
	Best-Effort Application Identification of DPI-Serviced Flows	5
	Features that Support Application-Level Filtering	5
	Best-Effort Application Determination	5
	APPID, ACL, and L-PDF Processing in Preconvergence Scenarios	5
	Prior to a Final or Best-Effort Application Identification	6
	Upon Best-Effort Application Identification	6
	While Application Identification Is on a Best-Effort Basis	6
	If a Flow Ends Before an Application Identification Is Made	6
	If a Flow Ends While Application Identification on a Best-Effort Basis	7
Part 2	Configuration	
Chapter 2	Configuration Tasks for Application Identification	11
	Defining an Application Identification	11
	Configuring APPID Rules	13
	Using Stateful Firewall Rules to Identify Data Sessions	15
	Configuring Application Profiles	17
	Configuring Application Groups	17
	Application Identification for Nested Applications	18
	Disabling Application Identification for Nested Applications	20
	Configuring Global APPID Properties	20
	Configuring APPID Support for Heuristics	22
	Configuring APPID Support for Unidirectional Traffic	22
	Configuring Automatic Download of Application Package Updates	23

	Tracing APPID Operations	24
	Configuring the APPID Log Filename	25
	Configuring the Number and Size of APPID Log Files	25
	Configuring Access to the Log File	25
	Configuring a Regular Expression for Lines to Be Logged	25
	Configuring the Tracing Flags	26
Chapter 3	Application Identification Example	27
	Examples: Configuring Application Identification Properties	27
Chapter 4	APPID Configuration Statements	29
	[edit services application-identification] Hierarchy Level	29
	address	31
	application (Defining)	32
	application (Including in Rule)	33
	application-group	33
	application-groups (Services Application Identification)	34
	application-system-cache-timeout	34
	applications (Services Application Identification)	35
	automatic	35
	bypass-traffic-on-exceeding-flow-limits	36
	chain-order	36
	context	37
	destination (Services)	37
	direction	38
	disable (APPID Application)	38
	disable (APPID Application Group)	38
	disable (APPID Port Mapping)	39
	disable-global-timeout-override	39
	download	40
	enable-asymmetric-traffic-processing	40
	enable-heuristics	41
	idle-timeout	41
	ignore-errors	42
	inactivity-non-tcp-timeout	42
	inactivity-tcp-timeout	43
	index	43
	index (Nested Applications)	44
	ip	44
	max-checked-bytes	45
	maximum-transactions	45
	member	46
	min-checked-bytes	46
	nested-application	47
	nested-application-settings	48
	no-application-identification	48
	no-application-system-cache	49
	no-clear-application-system-cache	49
	no-nested-application	50
	no-protocol-method	50

no-signature-based	51
order (Services Application Identification)	51
pattern	52
port-mapping	52
port-range	53
profile	53
protocol	54
rule (Configuring)	55
rule (Including in Rule Set)	56
rule-set (Services Application Identification)	56
services (Application Identification)	57
service-set (Services)	58
service-set-options	60
session-timeout (Interfaces)	60
session-timeout (Application Identification)	61
signature	61
signature-method-all-ports	62
source	62
support-uni-directional-traffic	63
traceoptions (Application Identification)	64
type	65
type-of-service	65
url	66

Part 3

Administration

Chapter 5

APPID Operational Mode Commands 69

clear services application-identification application-system-cache	70
clear services application-identification counter	71
request services application-identification application	72
request services application-identification group	74
show services application-identification application	75
show services application-identification application-system-cache	78
show services application-identification counter	80
show services application-identification group	83
show services application-identification version	85
show services flows	86

Part 4

Index

Index	95
-------	----

List of Tables

	About the Documentation ix
	Table 1: Notice Icons xi
	Table 2: Text and Syntax Conventions xi
Part 3	Administration
Chapter 5	APPID Operational Mode Commands 69
	Table 3: show services application-identification application Output Fields . . . 75
	Table 4: show application-identification application-system-cache Output Fields 78
	Table 5: show services application-identification counter Output Fields 80
	Table 6: show services application-identification group Output Fields 83
	Table 7: show services flows Output Fields 88

About the Documentation

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

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
- T Series
- MX 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 xi 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.

Table 2 on page xi 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

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

Convention	Description	Examples
<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)	Enclose optional keywords or variables.	stub <default-metric <i>metric</i>>;
(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</i> <i>string2</i> <i>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)	Enclose a variable for which you can substitute one or more values.	community name members [<i>community-ids</i>]
Indentation and braces ({ })	Identify 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		
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 send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release 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 JNASC 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: <https://www.juniper.net/alerts/>

- 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>.

PART 1

Overview

- [Application Identification on page 3](#)

CHAPTER 1

Application Identification

- [APPID Overview on page 3](#)
- [Best-Effort Application Identification of DPI-Serviced Flows on page 5](#)

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 *Junos OS Operational Mode Commands*.



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-aac1`

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 11](#)
- [Configuring APPID Rules on page 13](#)
- [Application Identification for Nested Applications on page 18](#)
- [Configuring Global APPID Properties on page 20](#)

- [Examples: Configuring Application Identification Properties on page 27](#)
- [\[edit services application-identification\] Hierarchy Level on page 29](#)

Best-Effort Application Identification of DPI-Serviced Flows

This topic describes the following information:

- [Features that Support Application-Level Filtering on page 5](#)
- [Best-Effort Application Determination on page 5](#)
- [APPID, AACL, and L-PDF Processing in Preconvergence Scenarios on page 5](#)

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 6](#)
- [Upon Best-Effort Application Identification on page 6](#)
- [While Application Identification Is on a Best-Effort Basis on page 6](#)

- [If a Flow Ends Before an Application Identification Is Made on page 6](#)
- [If a Flow Ends While Application Identification on a Best-Effort Basis on page 7](#)

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** 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*
- *Configuring Statistics Profiles*
- *aacl-fields*
- *aacl-statistics-profile*
- *rule*
- *services*
- *term*
- *then*

PART 2

Configuration

- [Configuration Tasks for Application Identification on page 11](#)
- [Application Identification Example on page 27](#)
- [APPID Configuration Statements on page 29](#)

CHAPTER 2

Configuration Tasks for Application Identification

- [Defining an Application Identification on page 11](#)
- [Configuring APPID Rules on page 13](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 15](#)
- [Configuring Application Profiles on page 17](#)
- [Configuring Application Groups on page 17](#)
- [Application Identification for Nested Applications on page 18](#)
- [Disabling Application Identification for Nested Applications on page 20](#)
- [Configuring Global APPID Properties on page 20](#)
- [Configuring APPID Support for Heuristics on page 22](#)
- [Configuring APPID Support for Unidirectional Traffic on page 22](#)
- [Configuring Automatic Download of Application Package Updates on page 23](#)
- [Tracing APPID Operations on page 24](#)

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 27](#).

Related Documentation

- [APPID Overview on page 3](#)
- [Configuring APPID Rules on page 13](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 15](#)
- [Configuring Application Profiles on page 17](#)
- [Configuring Application Groups on page 17](#)
- [Tracing APPID Operations on page 24](#)
- [\[edit services application-identification\] Hierarchy Level on page 29](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Using Stateful Firewall Rules to Identify Data Sessions on page 15](#)
- [Configuring Application Profiles on page 17](#)
- [Configuring Application Groups on page 17](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 3](#)
 - [Defining an Application Identification on page 11](#)
 - [Application Identification for Nested Applications on page 18](#)
 - [Configuring Global APPID Properties on page 20](#)
 - [Tracing APPID Operations on page 24](#)

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 13. 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 AACL rule set. For more information on service sets, see *Service Set Properties*.

- Related Documentation**
- [APPID Overview on page 3](#)
 - [Defining an Application Identification on page 11](#)
 - [Configuring Application Groups on page 17](#)
 - [Configuring Global APPID Properties on page 20](#)

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 11](#)
- [Configuring APPID Rules on page 13](#)
- [Configuring Application Profiles on page 17](#)
- [Configuring Global APPID Properties on page 20](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Disabling Application Identification for Nested Applications on page 20](#)

- [Configuring Global APPID Properties on page 20](#)
- [Tracing APPID Operations on page 24](#)

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 3](#)
- [Application Identification for Nested Applications on page 18](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Application Identification for Nested Applications on page 18](#)
- [Disabling Application Identification for Nested Applications on page 20](#)
- [Tracing APPID Operations on page 24](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Application Identification for Nested Applications on page 18](#)
- [Configuring Global APPID Properties on page 20](#)
- [Configuring APPID Support for Unidirectional Traffic on page 22](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Application Identification for Nested Applications on page 18](#)
- [Configuring Global APPID Properties on page 20](#)
- [Configuring APPID Support for Heuristics on page 22](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

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 *Junos OS System Log Messages Reference*.)
- 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 25](#)
- [Configuring the Number and Size of APPID Log Files on page 25](#)
- [Configuring Access to the Log File on page 25](#)

- [Configuring a Regular Expression for Lines to Be Logged on page 25](#)
- [Configuring the Tracing Flags on page 26](#)

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 3](#)
- [Defining an Application Identification on page 11](#)
- [Tracing APPID Operations on page 24](#)
- [Examples: Configuring Application Identification Properties on page 27](#)

CHAPTER 3

Application Identification Example

- [Examples: Configuring Application Identification Properties on page 27](#)

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;
  }
}
```

APPID Configuration Statements

- [\[edit services application-identification\] Hierarchy Level on page 29](#)

[\[edit services application-identification\] 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 11](#)
 - [Application Identification for Nested Applications on page 18](#)
 - [Configuring Global APPID Properties on page 20](#)

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 13](#)

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 11](#)

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 13

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	<i>group-name</i> —Unique identifier for the group. 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 Application Groups on page 17

application-groups (Services Application Identification)

Syntax	<code>application-groups { <i>application-group-name</i>; }</code>
Hierarchy Level	[edit services application-identification <code>application-group group-name</code>]
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 17

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 20

applications (Services Application Identification)

Syntax	<code>applications { <i>application-name</i>; }</code>
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 17

automatic

Syntax	<code>automatic { interval <i>hour</i>; start-time <i>time</i>; }</code>
Hierarchy Level	[edit services application-identification download]
Release Information	Statement introduced in Junos OS Release 9.5.
Description	Define automatic download properties.
Options	<p><i>interval hour</i>—Download interval in hours. The default is 24 and the range is from 1 through 168.</p> <p><i>start-time time</i>—Start-time value. The default is 0:00 and the range is from 0:00 through 24:00.</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 Automatic Download of Application Package Updates on page 23

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</i> <i>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 18

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 18

destination (Services)

Syntax	<pre>destination { ip address</prefix-length>; port-range { tcp [ports-and-port-ranges]; udp [ports-and-port-ranges]; } }</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 13

direction

Syntax	direction (any client-to-server server-to-client) ;
Hierarchy Level	[edit services application-identification nested-application name signature name member name]
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 18.

disable (APPID Application)

Syntax	disable;
Hierarchy Level	[edit services application-identification application application-name]
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 11

disable (APPID Application Group)

Syntax	disable;
Hierarchy Level	[edit services application-identification application-group group-name]
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 17

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 11

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 11

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 23

enable-asymmetric-traffic-processing

Syntax	<pre>enable-asymmetric-traffic-processing;</pre>
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	Enables 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 22


enable-heuristics

Syntax	enable-heuristics;
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	Enables 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 22

idle-timeout

Syntax	idle-timeout <i>seconds</i> ;
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	<i>seconds</i> —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 3 • Defining an Application Identification on page 11

ignore-errors

Syntax	ignore-errors <alg> <tcp>;
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	<p>alg—Mediate ALG behavior that results in dropping malformed packets or random packets when the software is unable to allocate resources.</p> <p>tcp—Prevent software from dropping packets that fail TCP SYN checks.</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">• Defining an Application Identification on page 11

inactivity-non-tcp-timeout

Syntax	inactivity-non-tcp-timeout <i>seconds</i> ;
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	<p>seconds—Timeout period.</p> <p>Range: 4 through 86,400</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">• Defining an Application Identification on page 11

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	<p><i>seconds</i>—Timeout period.</p> <p>Range: 4 through 86,400</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"> • Defining an Application Identification on page 11

index

Syntax	<code>index <i>number</i>;</code>
Hierarchy Level	<p>[edit services application-identification application <i>application-name</i>],</p> <p>[edit services application-identification application-group <i>group-name</i>]</p>
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	<p><i>number</i>—Index number; must be a unique, unsigned value.</p> <p>Range: 0 through 65535</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"> • Defining an Application Identification on page 11 • Configuring Application Groups on page 17

index (Nested Applications)

Syntax	<code>index <i>number</i>;</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	<i>number</i> —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 18.

ip

Syntax	<code>ip <i>address</i> </<i>prefix-length</i>>;</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	<i>address</i> </ <i>prefix-length</i> >—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 13

max-checked-bytes

Syntax	max-checked-bytes <i>bytes</i> ;
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 20

maximum-transactions

Syntax	maximum-transactions <i>number</i> ;
Hierarchy Level	[edit services application-identification <i>nested-application name signature name</i>]
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 18

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 18

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 20

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 18

nested-application-settings

Syntax	nested-application-settings { no-application-system-cache; no-nested-application; }
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Configure nested application options for application identification services.
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 18.

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 20

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 20 • Application Identification for Nested Applications on page 18.

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 20

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 18

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 20

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 20

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 13 • Application Identification for Nested Applications on page 18

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 18

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 application-name]
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 11

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 11 • Configuring APPID Rules on page 13

profile

Syntax	<pre>profile <i>profile-name</i> { rule-set <i>rule-set-name</i>; }</pre>
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	<i>profile-name</i> —Identifier for application profile. The remaining statement is 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 Application Profiles on page 17

protocol

Syntax	<code>protocol <i>protocol</i>;</code>
Hierarchy Level	[edit services application-identification nested-application name]
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	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 18

rule (Configuring)

```
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 13](#)

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 13

rule-set (Services Application Identification)

Syntax	<code>rule-set rule-set-name { rule application-rule-name; }</code>
Hierarchy Level	[edit services application-identification], [edit services application-identification profile profile-name]
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 13

services (Application Identification)

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

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 {
            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;
            support-uni-directional-traffic;
        }
        snmp-trap-thresholds {
            flows high high-threshold | low low-threshold;
            nat-address-port high-threshold | low low-threshold;
        }
        (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. The pgcp-rules and pgcp-rule-sets options were added in Release 8.4. The server-set-options option was added in Release 10.1. The ptsp-rules and ptsp-rule-sets options were added in Release 10.2. The softwire-rules and softwire-rule-sets options were added in Release 10.4.
Description	Define the service set.
Options	<p>service-set-name—Identifies the service set.</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"> <i>Service Set Properties</i>

service-set-options

Syntax	<pre>service-set-options { bypass-traffic-on-exceeding-flow-limits; bypass-traffic-on-pic-failure; enable-asymmetric-traffic-processing; support-uni-directional-traffic; header-integrity-check }</pre>
Hierarchy Level	[edit services service-set]
Release Information	Statement introduced in Junos OS Release 10.1. The enable-asymmetric-traffic-processing and the support-uni-directional-traffic options were added in Release 11.2.
Description	Specify the service set options to apply to a service set.
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">• <i>Configuring Service Sets to be Applied to Services Interfaces</i>• Configuring APPID Support for Unidirectional Traffic on page 22

session-timeout (Interfaces)

Syntax	<pre>session-timeout <i>seconds</i>;</pre>
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	seconds —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 11

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	<p><i>seconds</i>—Duration of session.</p> <p>Default: 3600</p> <p>Range: 1 through 604,800</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"> • Defining an Application Identification on page 11

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 <i>name</i>]
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	<p><i>name</i>—Name of the signature definition.</p> <p>The remaining statements are described 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 18

signature-method-all-ports

Syntax	signature-method-all-ports
Hierarchy Level	[edit services application-identification]
Release Information	Statement introduced in Junos OS Release 11.2.
Description	<p>Runs 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 20

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 13

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	Enables 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 22

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

Related Documentation

- [Tracing APPID Operations on page 24](#)

type

Syntax type *type*;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description Define type of application, such as HTTP or FTP.

Options **type**—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

- [Defining an Application Identification on page 11](#)
- [Application Identification for Nested Applications on page 18](#)

type-of-service

Syntax type-of-service *service-type*;

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

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 The following **service-type** options are available:

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

Related Documentation

- [Defining an Application Identification on page 11](#)

url

Syntax	<code>url url;</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	<code>url</code> —Download URL.
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 23

PART 3

Administration

- [APPID Operational Mode Commands on page 69](#)

CHAPTER 5

APPID Operational Mode Commands


[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 78

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 80

request services application-identification application

Syntax	<code>request services application-identification application copy <i>predefined-application-name</i></code> <code>request services application-identification application [disable enable]</code> <code><i>predefined-application-name</i> <no-commit></code>
Release Information	Command introduced in Release 11.4 of Junos OS.
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> <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> <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 75](#)

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 group

Syntax	<code>request services application-identification group [copy disable enable] predefined-application-group-name</code>
Release Information	Command introduced in Release 11.4 of Junos OS.
Description	Copy, disable, or enable a predefined application signature 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 83
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 copy junos:SYBASE  
group 1040 copied successfully.
```


show services application-identification application

Syntax	show services application-identification application [detail <i>application-name</i>] summary]
Release Information	Command introduced in Release 11.4 of Junos OS.
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 72
List of Sample Output	show services application-identification application summary on page 76 show services application-identification application detail on page 76
Output Fields	Table 3 on page 75 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 3: 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 3: 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 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 78](#)

Output Fields [Table 4 on page 78](#) lists the output fields for the **command-name** command. Output fields are listed in the approximate order in which they appear.

Table 4: 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 81 show services application-identification counter on page 81
Output Fields	Table 5 on page 80 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 5: 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 5: 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 Release 11.4 of Junos OS.
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 74
List of Sample Output	show services application-identification group summary on page 83 show services application-identification group detail on page 84
Output Fields	Table 6 on page 83 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 6: 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
my:enterprise                Disabled  ID
                             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:BADOO;
    junos:BEBO;
    junos:BLOGGER-POST;
    junos:BLOGSPOT-POST;
```

show services application-identification version

Syntax	show services application-identification version
Release Information	Command introduced in Release 10.2 of Junos OS.
Description	Display the Junos OS application package version.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• <i>request services application-identification download</i>
List of Sample Output	show services application-identification version on page 85

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**. 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.

Required Privilege Level view

Related Documentation

- *clear services flows*

List of Sample Output [show services flows on page 89](#)
[show services flows all on page 89](#)
[show services flows brief on page 90](#)
[show services flows extensive on page 90](#)
[show services flows application-protocol on page 90](#)
[show services flows count on page 90](#)
[show services flows destination port on page 90](#)
[show services flows destination prefix on page 91](#)
[show services flows interface on page 91](#)
[show services flows protocol on page 91](#)
[show services flows service-set on page 91](#)
[show services flows source port on page 91](#)
[show services flows source prefix on page 91](#)

Output Fields [Table 7 on page 88](#) lists the output fields for the **show services flows** command. Output fields are listed in the approximate order in which they appear.

Table 7: 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 7: 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@router> 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@router> 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@router> 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 O     137
ICMP      10.2.2.2      -> 10.1.1.2      Forward I     132

```

show services flows interface

```

user@router> 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 O      5
ICMP      10.1.1.2      -> 10.2.2.2      Forward I     162
ICMP      10.2.2.2      -> 10.1.1.2      Forward O     162
ICMP      10.2.2.2      -> 10.1.1.2      Forward I     157
ICMP      10.1.1.2      -> 10.2.2.2      Forward O     157

```

show services flows protocol

```

user@router> 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 O     202
ICMP      10.2.2.2      -> 10.1.1.2      Forward I     197
ICMP      10.1.1.2      -> 10.2.2.2      Forward O     197

```

show services flows service-set

```

user@router> 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 O      5
ICMP      10.1.1.2      -> 10.2.2.2      Forward I     220
ICMP      10.2.2.2      -> 10.1.1.2      Forward O     220
ICMP      10.2.2.2      -> 10.1.1.2      Forward I     215
ICMP      10.1.1.2      -> 10.2.2.2      Forward O     215

```

show services flows source port

```

user@router> 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 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 flows source prefix

```

user@router> 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

PART 4

Index

- [Index on page 95](#)

Index

Symbols

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

A

AACL	
best-effort application identification.....	5
address statement	
APPID	
usage guidelines.....	13
application rule.....	31
APPID	
best-effort application identification.....	5
example configuration.....	27
application statement.....	32, 33
APPID	
usage guidelines.....	11
application-group statement.....	33
APPID	
usage guidelines.....	17
application-groups statement.....	34
APPID	
usage guidelines.....	17
application-system-cache-timeout statement.....	34
APPID	
usage guidelines.....	20
applications statement	
APPID	
usage guidelines.....	17
application identification.....	35
asymmetrical routing support	
APPID.....	22
automatic statement.....	35
APPID	
usage guidelines.....	23

B

best-effort application identification.....	5
braces, in configuration statements.....	xii
brackets	
angle, in syntax descriptions.....	xii
square, in configuration statements.....	xii
bypass-traffic-on-exceeding-flow-limits	
statement.....	36

C

chain-order statement	
nested applications.....	36
clear services application-identification	
application-system-cache command.....	70
clear services application-identification counter	
command.....	71
comments, in configuration statements.....	xii
context statement	
nested applications.....	37
conventions	
text and syntax.....	xi
curly braces, in configuration statements.....	xii
customer support.....	xiii
contacting JTAC.....	xiii

D

data session identification	
APPID.....	15
destination statement	
APPID	
usage guidelines.....	13
application identification rule.....	37
direction statement	
nested applications.....	38
disable statement	
APPID	
usage guidelines.....	11, 17
application.....	38
application group.....	38
port mapping.....	39
disable-global-timeout-override statement.....	39
usage guidelines.....	11
documentation	
comments on.....	xiii
download statement	
APPID.....	40
usage guidelines.....	23

E

enable-asymmetric-traffic-processing statement.....	40
enable-heuristics statement.....	41
usage guidelines.....	22
encrypted traffic identification APPID.....	22
event policy	
all (tracing flag) APPID.....	26

F

font conventions.....	xi
-----------------------	----

H

heuristics support APPID.....	22
-------------------------------	----

I

idle-timeout statement.....	41
APPID	
usage guidelines.....	11
ignore-errors statement.....	42
usage guidelines.....	15
inactivity-non-tcp-timeout statement.....	42
usage guidelines.....	11
inactivity-tcp-timeout statement.....	43
usage guidelines.....	11
index statement.....	43
APPID	
usage guidelines.....	11, 17
nested applications.....	44
ip statement	
APPID	
usage guidelines.....	13
application identification.....	44

L

L-PDF	
best-effort application identification.....	5
log output	
APPID.....	25

M

manuals	
comments on.....	xiii
max-checked-bytes statement.....	45
APPID	
usage guidelines.....	20

maximum-transactions statement	
nested applications.....	45
member statement	
nested applications.....	46
min-checked-bytes statement.....	46
APPID	
usage guidelines.....	20

N

nested-application statement	
APPID.....	47
usage guidelines.....	18
nested-application-settings statement	
APPID.....	48
no-application-identification statement.....	48
APPID	
usage guidelines.....	20
no-application-system-cache statement.....	49
APPID	
usage guidelines.....	20
no-clear-application-system-cache statement.....	49
APPID	
usage guidelines.....	20
no-nested-application statement.....	50
usage guidelines.....	20
no-protocol-method statement.....	50
APPID	
usage guidelines.....	20
no-signature-based statement.....	51
APPID	
usage guidelines.....	20

O

order statement.....	51
APPID	
usage guidelines.....	13

P

parentheses, in syntax descriptions.....	xii
pattern statement	
nested applications.....	52
port-mapping statement.....	52
port-range statement.....	53
APPID	
usage guidelines.....	13

profile statement	
APPID	
usage guidelines.....	17
application identification.....	53
protocol statement	
nested applications.....	54

R

request services application-identification	
application command.....	72
request services application-identification group	
command.....	74
rule statement	
APPID	
usage guidelines.....	13
rule-set statement	
APPID	
usage guidelines.....	13
application identification.....	56

S

senable-asymmetric-traffic-processing statement	
usage guidelines.....	22
service-set statement.....	58
services statement	
APPID	
usage guidelines.....	29
session-timeout statement.....	60, 61
usage guidelines.....	11
show application-identification	
application-system-cache command.....	78
show services application-identification application	
command.....	75
show services application-identification counter	
command.....	80
show services application-identification group	
command.....	83
show services application-identification	
version.....	85
show services flows command.....	86
signature statement	
nested applications.....	61
signature-method-all-ports statement.....	62
AAPID	
usage guidelines.....	20
source statement	
APPID	
usage guidelines.....	13
application identification rule.....	62

stateful firewall use with APPID.....	15
support, technical See technical support	
support-uni-directional-traffic statement.....	63
usage guidelines.....	22
syntax conventions.....	xi

T

technical support	
contacting JTAC.....	xiii
traceoptions statement	
application identification.....	64
tracing flags	
event policy	
all.....	26
tracing operations	
APPID.....	24
type statement.....	65
APPID	
usage guidelines.....	11
type-of-service statement.....	65
APPID	
usage guidelines.....	11

U

unidirectional traffic support	
APPID.....	22
url statement.....	66
APPID	
usage guidelines.....	23

