Application Identification

Published: 2012-02-28

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

Application Identification

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

List of Tables

About the Documentation

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.xsl;
       }
     }
   }
   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.xsl; }
   ```

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 *Junos OS 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
Italic text like this	• Introduces important new terms. • Identifies book names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS System Basics Configuration Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	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** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; interface names; configuration hierarchy levels; or labels on routing platform components.	• 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** *metric*>**;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** (***string1*** **|** ***string2*** **|** ***string3***)
# (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 [** ***community-ids*** **]**

Indention and braces ({ })	Identify a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web 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

CHAPTER 1

Application Identification

APPID Overview

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 System Basics and Services Command Reference*.

PART 2

Configuration

CHAPTER 2

Configuration Tasks for Application Identification

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 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 21.

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;
}
```

For a configuration example, see "Examples: Configuring Application Identification Properties" on page 21.

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 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.

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 9. 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 Dynamic Application Awareness include the APPID and IDP profiles and the AACL rule set. For more information on service sets, see Service Set Properties.

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.

For a configuration example, see "Examples: Configuring Application Identification Properties" on page 21.

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.

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

- Application Identification for Nested Applications on page 13

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:

```
[edit services]
application-identification {
  application-system-cache-timeout seconds;
  max-checked-bytes bytes;
  min-checked-bytes bytes;
  nested-applicationname
  nested-application-settings
  no-application-identification
  no-application-system-cache;
  no-clear-application-system-cache;
  no-protocol-method;
  no-signature-based;
}
```

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 seenested-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.

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.

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-asymmetic-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.

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.

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 18
- Configuring the Number and Size of APPID Log Files on page 19
- Configuring Access to the Log File on page 19
- Configuring a Regular Expression for Lines to Be Logged on page 19
- Configuring the Tracing Flags on page 19

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.

CHAPTER 3

Application Identification Example

Examples: Configuring Application Identification Properties

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

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

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

The following examples show application group configuration:

```
[edit services application-identification]
```

```
application-group junos:peer-to-peer {
  index 5;
  application-groups {
    junos:chat;
    junos:file-sharing;
    junos:voip;
  }
}
```

```
[edit services application-identification]
application-group junos:voip {
  index 14;
  applications {
    junos:h225ras;
    junos:h225sgn;
    junos:mgcp;
    junos:sip;
  }
}
```

The following examples show application identification for nested application configuration:

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

CHAPTER 4

APPID Configuration Statements

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 9

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 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 7

application (Including in Rule)

Syntax application *application-name*;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description Identify the application for inclusion in a rule.

Options ***application-name***—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
- Configuring APPID Rules on page 9

application-group

Syntax

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

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 ***group-name***—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
- Configuring Application Groups on page 12

application-groups

Syntax

```
application-groups {
  application-group-name;
}
```

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 application groups for inclusion in a larger application group. An ***application-group-name*** statement is mandatory.

Options ***application-group-name***—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

- Configuring Application Groups on page 12

application-system-cache-timeout

Syntax application-system-cache-timeout *seconds*;

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 ***seconds***— 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

- Configuring Global APPID Properties on page 15

applications

Syntax

```
applications {
  application-name;
}
```

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 ***application-name***—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

- Configuring Application Groups on page 12

automatic

Syntax

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

Hierarchy Level [edit services application-identification download]

Release Information Statement introduced in Junos OS Release 9.5.

Description Define automatic download properties.

Options **interval** ***hour***—Download interval in hours. The default is **24** and the range is from 1 through 168.

start-time ***time***—Start-time value. The default is **0:00** and the range is from 0:00 through 24:00.

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

Related Documentation

- Configuring Automatic Download of Application Package Updates on page 17

chain-order

Syntax chain-order;

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

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
- Application Identification for Nested Applications on page 13

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
- Application Identification for Nested Applications on page 13

destination

Syntax

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

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
- Configuring APPID Rules on page 9

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 ***direction***—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
- Application Identification for Nested Applications on page 13.

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
- Defining an Application Identification on page 7

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
- Configuring Application Groups on page 12

disable (APPID Port Mapping)

Syntax disable;

Hierarchy Level [edit services application-identification application *application-name* 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
- Defining an Application Identification on page 7

disable-global-timeout-override

Syntax disable-global-timeout-override;

Hierarchy Level [edit interfaces *interface-name* 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
- Defining an Application Identification on page 7

download

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

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
- Configuring Automatic Download of Application Package Updates on page 17

enable-asymmetric-traffic-processing

Syntax enable-asymmetric-traffic-processing;

Hierarchy Level [edit services service-set *service-set-name* 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
- Configuring APPID Support for Unidirectional Traffic on page 16

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
- Configuring APPID Support for Heuristics on page 16

idle-timeout

Syntax idle-timeout *seconds*;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description Define idle timeout for an application in seconds. When the timeout period expires, the session ends if no packets have been received.

Options ***seconds***—Idle timeout period.
Default: 30
Range: 1 through 604,800

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

Related Documentation
- Defining an Application Identification on page 7

ignore-errors

Syntax ignore-errors <alg> <tcp>;

Hierarchy Level [edit interfaces *interface-name* services-options]

Release Information Statement introduced in Junos OS Release 10.1.

Description Define settings for minimizing TCP packet drops during stateful firewall processing.

Options **alg**—Mediate ALG behavior that results in dropping malformed packets or random packets when the software is unable to allocate resources.

tcp—Prevent software from dropping packets that fail TCP SYN checks.

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

Related Documentation
- Defining an Application Identification on page 7

inactivity-non-tcp-timeout

Syntax inactivity-non-tcp-timeout *seconds*;

Hierarchy Level [edit interfaces *interface-name* 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 ***seconds***—Timeout period.
Range: 4 through 86,400

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

Related Documentation
- Defining an Application Identification on page 7

inactivity-tcp-timeout

Syntax inactivity-tcp-timeout *seconds*;

Hierarchy Level [edit interfaces *interface-name* 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 ***seconds***—Timeout period.
Range: 4 through 86,400

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

Related Documentation
- Defining an Application Identification on page 7

index

Syntax index *number*;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description Assign an application or application-group index number. This is a mandatory value.

Options *number*—Index number; must be a unique, unsigned value.
Range: 0 through 65535

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

Related Documentation
- Defining an Application Identification on page 7
- Configuring Application Groups on page 12

index (Nested Applications)

Syntax index *number*;

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

Release Information Statement introduced in Junos OS Release 10.2.

Description Set a number that is a one-to-one mapping to the application name. The application name is used to ensure that each signature definition is unique.

Options *number*—Numeric value associated with an application name. The index range for predefined applications is from 1 through 32767. The index range for custom applications and custom nested applications is from 32768 through 65534.

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

Related Documentation
- Application Identification for Nested Applications on page 13.

ip

Syntax ip *address</prefix-length>*;

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 ***address</prefix-length>***—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
- Configuring APPID Rules on page 9

max-checked-bytes

Syntax max-checked-bytes *bytes*;

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 ***bytes***—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
- Configuring Global APPID Properties on page 15

maximum-transactions

Syntax maximum-transactions *number*;

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

Release Information Statement introduced in Junos OS Release 10.2.

Description Set the maximum number of transactions required before a match is made.

Options ***number***—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
- Application Identification for Nested Applications on page 13

member

Syntax member *name*;

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

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 ***name***—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
- Application Identification for Nested Applications on page 13

min-checked-bytes

Syntax min-checked-bytes *bytes*;

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 ***bytes***—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
- Configuring Global APPID Properties on page 15

nested-application

Syntax

```
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;
}
```

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 *name*—Name of nested application.

The remaining statements are explained separately.

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

Related Documentation

- Application Identification for Nested Applications on page 13

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
- Application Identification for Nested Applications on page 13.

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
- Configuring Global APPID Properties on page 15

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
- Configuring Global APPID Properties on page 15
- Application Identification for Nested Applications on page 13.

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
- Configuring Global APPID Properties on page 15

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
- Application Identification for Nested Applications on page 13

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
- Configuring Global APPID Properties on page 15

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
- Configuring Global APPID Properties on page 15

order

Syntax order *number*;

Hierarchy Level [edit services application-identification nested-application *name* signature *name* member *name*]
[edit services application-identification rule *rule-name* address]

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 *number*—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
- Configuring APPID Rules on page 9
- Application Identification for Nested Applications on page 13

pattern

Syntax pattern *dfa-pattern*;

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 ***dfa-pattern***—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
- Application Identification for Nested Applications on page 13

port-mapping

Syntax
```
port-mapping {
  disable;
  port-range {
    tcp [ ports-and-port-ranges ];
    udp [ ports-and-port-ranges ];
  }
}
```

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
- Defining an Application Identification on page 7

port-range

Syntax

```
port-range {
  tcp [ ports-and-port-ranges ];
  udp [ ports-and-port-ranges ];
}
```

Hierarchy Level [edit services application-identification application *application-name* port-mapping],
[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 TCP and UDP port numbers or numeric ranges. For port-mapping configurations, this entry is required if the parent node exists.

Options ***ports-and-port-ranges***—Individual port numbers, numeric port ranges, or both. Separate the values with spaces. The format for numeric port ranges is *minimum-value–maximum-value*.

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 7
- Configuring APPID Rules on page 9

profile

Syntax

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

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 ***profile-name***—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

- Configuring Application Profiles on page 12

protocol

Syntax protocol *protocol*;

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 ***protocol***—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

- Application Identification for Nested Applications on page 13

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 9

rule (Including in Rule Set)

Syntax rule *rule-name*;

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 ***rule-name***—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
- Configuring APPID Rules on page 9

rule-set

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

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 ***rule-set-name***—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
- Configuring APPID Rules on page 9

services

Syntax services application-identification { ... }

Hierarchy Level [edit]

Release Information **services** statement introduced before Junos OS Release 7.4.
application-identification statement introduced in Junos OS Release 9.5.

Description Define the services to be applied to traffic.

Options **application-identification**—The values configured for application-identification properties.

The 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
- Application Identification

session-timeout (Interfaces)

Syntax session-timeout *seconds*;

Hierarchy Level [edit interfaces *interface-name* 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
- Defining an Application Identification on page 7

session-timeout (Application Identification)

Syntax session-timeout *seconds*;

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

Release Information Statement introduced in Junos OS Release 9.5.

Description Define session lifetime for the specified application in seconds.

Options *seconds*—Duration of session.
Default: 3600
Range: 1 through 604,800

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

Related Documentation
- Defining an Application Identification on page 7

signature

Syntax
```
signature name {
  chain-order;
  maximum-transactions number;
  member name {
    context value;
    direction (any | client-to-server | server-to-client);
    pattern dfa-pattern;
  }
  order number;
}
```

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

Release Information Statement introduced in Junos OS Release 10.2.

Description Identify the name of the custom nested application signature definition. The name must be unique with a maximum length of 32 characters.

Options *name*—Name of the signature definition.

The remaining statements are described separately.

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

Related Documentation
- Application Identification for Nested Applications on page 13

source

Syntax

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

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

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
- Configuring APPID Rules on page 9

support-uni-directional-traffic

Syntax support-uni-directional-traffic;

Hierarchy Level [edit services service-set *service-set-name* 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
- Configuring APPID Support for Unidirectional Traffic on page 16

traceoptions

Syntax

```
traceoptions {
  file filename <files number> <match regex> <size size> <world-readable |
    no-world-readable>;
  flag flag;
  no-remote-trace;
}
```

Hierarchy Level [edit services application-identification]

Release Information Statement introduced in Junos OS Release 9.5.

Description Configure application identification tracing options.

To specify more than one tracing operation, include multiple **flag** statements.

Options

file *filename*—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**.

files *number*—(Optional) Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed ***trace-file*.0**, then ***trace-file*.1**, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

Range: 2 through 1000 files

Default: 2 files

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

flag—Tracing operation to perform. **all** is the only valid completion.

- **all**—Trace all events.

match *regex*—(Optional) Regular expression for lines to be logged.

no-world-readable—(Optional) Disallow any user to read the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named ***trace-file*** reaches this size, it is renamed ***trace-file*.0**. When the ***trace-file*** again reaches its maximum size, ***trace-file*.0** is renamed ***trace-file*.1** and ***trace-file*** is renamed ***trace-file*.0**. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

Syntax: ***x*k** to specify KB, ***x*m** to specify MB, or ***x*g** to specify GB

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

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

world-readable—(Optional) Allow any user to read the log file.

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 18

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 7
- Application Identification for Nested Applications on page 13

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 7

url

Syntax url *url*;

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 ***url***—Download URL.

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

Related Documentation

- Configuring Automatic Download of Application Package Updates on page 17

PART 3

Administration

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

- show services application-identification application-system-cache on page 60

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
- show services application-identification counter on page 62

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 60

Output Fields Table 3 on page 60 lists the output fields for the **command-name** command. Output fields are listed in the approximate order in which they appear.

Table 3: 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 *interface-name*>

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 *interface-name***—Displays the services interfaces to query.

Required Privilege Level view

List of Sample Output show services application-identification counter on page 63
show services application-identification counter on page 63

Output Fields Table 4 on page 62 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 4: 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 4: 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 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 **rsp*number***. On J Series routers, ***interface-name*** is **ms-*pim*/0/*port***.

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

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

- ***number***—Numeric protocol value from 0 to 255
- **ah**—IPsec Authentication Header protocol
- **egp**—An exterior gateway protocol
- **esp**—IPsec Encapsulating Security Payload protocol
- **gre**—A generic routing encapsulation protocol
- **icmp**—Internet Control Message Protocol
- **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 68
show services flows all on page 68
show services flows brief on page 69
show services flows extensive on page 69
show services flows application-protocol on page 69
show services flows count on page 69
show services flows destination port on page 69
show services flows destination prefix on page 69
show services flows interface on page 70
show services flows protocol on page 70
show services flows service-set on page 70
show services flows source port on page 70
show services flows source prefix on page 70

Output Fields Table 5 on page 67 lists the output fields for the **show services flows** command. Output fields are listed in the approximate order in which they appear.

Table 5: 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 5: 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 ***source-prefix:port***. 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: • **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

Symbols

A

B

C

D

E