Junos® OS

Intrusion Detection and Prevention Feature Guide for Security Devices

Modified: 2018-06-18

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

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

Junos® OS Intrusion Detection and Prevention Feature Guide for Security Devices

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 https://www.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

List of Figures

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 https://www.juniper.net/documentation/.

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 https://www.juniper.net/books.

Supported Platforms

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

- vSRX
- SRX 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 CLI Explorer.

Documentation Conventions

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

Table 1: Notice Icons

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

Table 2 on page xxiv 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`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User 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; 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)	Encloses optional keywords or variables.	**stub <default-metric *metric*>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*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)	Encloses a variable for which you can substitute one or more values.	**community name members [*community-ids*]**

Indention and braces ({ })	Identifies 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.	

GUI Conventions

Table 2: Text and Syntax Conventions (continued)

Convention	Description	Examples
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

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

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

Requesting Technical Support

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

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://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: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: https://www.juniper.net/cm/

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

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 https://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 https://www.juniper.net/support/requesting-support.html.

CHAPTER 1

Overview

Intrusion Detection and Prevention Overview

Supported Platforms SRX Series, vSRX

Intrusion detection is the process of monitoring the events occurring in your network and analyzing them for signs of possible incidents, violations, or imminent threats to your security policies. Intrusion prevention is the process of performing intrusion detection and then stopping the detected incidents. These security measures are available as intrusion detection systems (IDS) and intrusion prevention systems (IPS), which become part of your network to detect and stop potential incidents.

For more information, see the following topics:

- Understanding Intrusion Detection and Prevention for SRX Series on page 27
- Understanding IDP Inline Tap Mode on page 28
- Example: Configuring IDP Inline Tap Mode on page 29

Understanding Intrusion Detection and Prevention for SRX Series

An Intrusion Detection and Prevention (IDP) policy lets you selectively enforce various attack detection and prevention techniques on the network traffic passing through your SRX Series. The SRX Series offer the same set of IDP signatures that are available on Juniper Networks IDP Series Intrusion Detection and Prevention Appliances to secure networks against attacks. The basic IDP configuration involves the following tasks:

- Download and install the IDP license.
- Download and install the signature database—You must download and install the IDP signature database. The signature databases are available as a security package on the Juniper Networks website. This database includes attack object and attack object groups that you can use in IDP policies to match traffic against known attacks.
- Configure recommended policy as the IDP policy—Juniper Networks provides predefined policy templates to use as a starting point for creating your own policies. Each template is a set of rules of a specific rulebase type that you can copy and then update according to your requirements.

To get started, we recommend you use the predefined policy named “Recommended”.

- Enable a security policy for IDP inspection—For transit traffic to pass through IDP inspection, you configure a security policy and enable IDP application services on all traffic that you want to inspect.

SRX Series Services Gateways can be deployed in inline tap mode and sniffer mode (only on SRX5400, SRX5600, and SRX5800 devices). The sniffer mode is not supported on SRX300, SRX320, SRX340, SRX345, and SRX550HM devices.

NOTE: **Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, inline tap mode is not supported on SRX Series devices.**

Sniffer mode is supported only on SRX5400, SRX5600, and SRX5800 devices. You can use the sniffer mode of IDP deployment by configuring the interfaces in promiscuous mode and manipulating the traffic and flow setup with routing.

On SRX5400, SRX5600, and SRX5800 devices, in sniffer mode, ingress and egress interfaces work with flow showing both source and destination interface as egress interface.

As a workaround, in sniffer mode, use the tagged interfaces. Hence, the same interface names are displayed in the logs. For example, the ge-0/0/2.0 as ingress (sniff) and the ge-0/0/2.100 as egress interfaces are displayed in the logs to show the source interface as ge-0/0/2.100.

```
set interfaces ge-0/0/2 promiscuous-mode
set interfaces ge-0/0/2 vlan-tagging
set interfaces ge-0/0/2 unit 0 vlan-id 0
set interfaces ge-0/0/2 unit 100 vlan-id 100
```

See Also

- *Example: Configuring Intrusion Detection and Prevention for SRX Series*

Understanding IDP Inline Tap Mode

NOTE: **Starting in Junos OS Release 15.1X49-D10, IDP inline tap mode is not supported on SRX Series devices.**

The main purpose of inline tap mode is to provide best case deep inspection analysis of traffic while maintaining over all performance and stability of the device. The inline tap feature provides passive, inline detection of application layer threats for traffic matching security policies which have the IDP application service enabled. When a device is in inline tap mode, packets pass through firewall inspection and are also copied to the independent IDP module. This allows the packets to get to the next service module without waiting for IDP processing results. By doing this, when the traffic input is beyond the IDP throughput limit, the device can still sustain processing as long as it does not go beyond the modules limits, such as with the firewall. If the IDP process fails, all other features of the device will continue to function normally. Once the IDP process recovers, it will resume processing packets for inspection. Since inline tap mode puts IDP in a passive mode for monitoring, preventative actions such as session close, drop, and mark diffserv are deferred. The action drop packet is ignored.

Inline tap mode can only be configured if the forwarding process mode is set to maximize IDP sessions, which ensures stability and resiliency for firewall services. You also do not need a separate tap or span port to use inline tap mode.

NOTE: **You must restart the device when switching to inline tap mode or back to regular mode.**

Example: Configuring IDP Inline Tap Mode

This example shows how to configure a device for inline tap mode.

Requirements

Before you begin, review the inline tap mode feature. See "Understanding IDP Inline Tap Mode" on page 28.

NOTE: **Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, IDP inline tap mode is not supported on SRX Series devices.**

Overview

The inline tap mode feature provides passive, inline detection of Application Layer threats for traffic matching security policies that have the IDP application service enabled.

NOTE: **IDP inline tap mode does not require a separate tap or span port.**

Configuration

Step-by-Step Procedure

To configure a device for inline tap mode:

1. Set inline tap mode.

```
[edit]
user@host# set security forwarding-process application-services
  maximize-idp-sessions inline-tap
```

2. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

3. Restart the system from operational mode.

```
user@host> request system reboot
```

NOTE: When switching to inline tap mode or back to regular mode, you must restart the device.

4. If you want to switch the device back to regular mode, delete inline tap mode configuration.

   ```
   [edit security]
   user@host# delete forwarding-process application-services maximize-idp-sessions
     inline-tap
   ```

Verification

To verify that inline tap mode is enabled, enter the **show security idp status** command. The line item for the forwarding process mode shows **“Forwarding process mode: maximizing sessions (Inline-tap)”**.

CHAPTER 2

Downloading and Updating the IDP Signature Database

IDP Signature Database Overview

Supported Platforms SRX Series, vSRX

Signature-based IDP monitors packets in the Network and compares with pre-configured and pre-determined attack patterns known as signatures.

For more information, see the following topics:

- Understanding the IDP Signature Database on page 31
- Updating the IDP Signature Database Overview on page 32
- Example: Updating the Signature Database Automatically on page 33
- Updating the IDP Signature Database Manually Overview on page 34
- Example: Updating the IDP Signature Database Manually on page 35
- Example: Downloading and Installing the IDP Security Packages in Chassis Cluster Mode on page 38
- Understanding the IDP Signature Database Version on page 41
- Verifying the IDP Signature Database Version on page 41

Understanding the IDP Signature Database

The signature database is one of the major components of Intrusion Detection and Prevention (IDP). It contains definitions of different objects—such as attack objects, application signatures objects, and service objects—that are used in defining IDP policy rules. As a response to new vulnerabilities, Juniper Networks periodically provides a file containing attack database updates on the Juniper website. You can download this file to protect your network from new threats.

NOTE: IDP feature is enabled by default, no license is required. Custom attacks and custom attack groups in IDP policies can also be configured and installed even when a valid license and signature database are not installed on the device.

The IDP signature database is stored on the IDP enabled device and contains definitions of predefined attack objects and groups. These attack objects and groups are designed to detect known attack patterns and protocol anomalies within the network traffic. You can configure attack objects and groups as match conditions in IDP policy rules.

NOTE: You must install the IDP signature-database-update license key on your device for downloading and installing daily signature database updates provided by Juniper Networks. The IDP signature license key does not provide grace period support. For license details, see *Junos OS Feature License Keys*.

You can perform the following tasks to manage the IDP signature database:

1. Update the signature database—Download the attack database updates available on the Juniper Networks website. New attacks are discovered daily, so it is important to keep your signature database up to date.
2. Verify the signature database version—Each signature database has a different version number with the latest database having the highest number. You can use the CLI to display the signature database version number.
3. Update the protocol detector engine—You can download the protocol detector engine updates along with downloading the signature database. The IDP protocol detector contains Application Layer protocol decoders. The detector is coupled with the IDP policy and is updated together. It is always needed at policy update time, even if there is no change in the detector.
4. Schedule signature database updates—You can configure the IDP-enabled device to automatically update the signature database after a set interval.

Updating the IDP Signature Database Overview

Juniper Networks regularly updates the predefined attack database and makes it available on the Juniper Networks website. This database includes attack object groups that you can use in Intrusion Detection and Prevention (IDP) policies to match traffic against known attacks. Although you cannot create, edit, or delete predefined attack objects, you can use the CLI to update the list of attack objects that you can use in IDP policies.

To update the signature database, you download a security package from the Juniper Networks website. The security package consists of the following IDP components:

- Attack objects
- Attack object groups
- Application objects

- Updates to the IDP Detector Engine
- IDP Policy templates (Policy templates are downloaded independently. See "Understanding Predefined IDP Policy Templates" on page 51.)

By default, when you download the security package, you download the following components into a Staging folder in your device: the latest version of the complete attack object groups table, application objects table, and the updates to the IDP Detector Engine. Because the attack objects table is typically of a large size, by default the system downloads only updates to the attack objects table. However, you can download the complete attack objects table by using the **full-update** configuration option.

After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device.

After installing a security package, when you commit the configuration, all policies are checked for their syntax (not only the active policy). This checking is the same as a commit check. If an attack configured in any of the existing policies is removed from the new signature database that you download, the commit check fails.

When you update the IDP signature database, attacks configured in policies are not updated automatically. For example, suppose you configure a policy to include an attack **FTP:USER:ROOT** that is available in the signature database version 1200 on your system. Then, you download signature database version 1201, which no longer includes the attack **FTP:USER:ROOT**. Because an attack configured in your policy is missing from the newly downloaded database, the commit check in the CLI fails. To successfully commit your configuration, you must remove the attack (**FTP:USER:ROOT**) from your policy configuration.

CAUTION: IDP signature updates might fail if a new IDP policy load fails for any reason. When a new IDP policy load fails, the last known good IDP policy is loaded. Once the issue with the new policy load is resolved, and the new valid policy is active, signature updates will work properly.

See Also

- Understanding Predefined IDP Attack Objects and Object Groups on page 92

Example: Updating the Signature Database Automatically

This example shows how to download signature database updates automatically.

- Requirements on page 33
- Overview on page 34
- Configuration on page 34

Requirements

Before you begin, configure network interfaces.

Overview

Juniper Networks regularly updates the predefined attack database and makes it available as a security package on the Juniper Networks website. This database includes attack objects and attack object groups that you can use in IDP policies to match traffic against known attacks. You can configure your device to automatically download the signature database updates at specified intervals.

In this example, you download the security package with the complete table of attack objects and attack object groups every 48 hours, starting at 11:59 p.m. on December 10. You also enable an automatic download and update of the security package.

Configuration

Step-by-Step Procedure

To download and update the predefined attack objects:

1. Specify the URL for the security package.

   ```
   [edit]
   user@host# set security idp security-package url
     https://services.netscreen.com/cgi-bin/index.cgi
   ```

   NOTE: By default it will take URL as https://services.netscreen.com/cgi-bin/index.cgi.

2. Enable the automatic download and update of the security package.

   ```
   [edit]
   user@host# set security idp security-package automatic enable
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

Updating the IDP Signature Database Manually Overview

Juniper Networks regularly updates the predefined attack database and makes it available on the Juniper Networks website. This database includes attack object groups that you can use in Intrusion Detection and Prevention (IDP) policies to match traffic against known attacks. Although you cannot create, edit, or delete predefined attack objects, you can use the CLI to update the list of attack objects that you can use in IDP policies. After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device.

Example: Updating the IDP Signature Database Manually

This example shows how to update the IDP signature database manually.

- Requirements on page 35
- Overview on page 35
- Configuration on page 35
- Verification on page 38

Requirements

Before you begin, configure network interfaces.

Overview

Juniper Networks regularly updates the predefined attack database and makes it available as a security package on the Juniper Networks website. This database includes attack object and attack object groups that you can use in IDP policies to match traffic against known attacks.

In this example, you download the security package with the complete table of attack objects and attack object groups. Once the installation is completed, the attack objects and attack object groups are available in the CLI under the predefined-attack-groups and predefined-attacks configuration statements at the [edit security idp idp-policy] hierarchy level. You create a policy and specify the new policy as the active policy. You also download only the updates that Juniper Networks has recently uploaded and then update the attack database, the running policy, and the detector with these new updates.

Configuration

CLI Quick Configuration

CLI quick configuration is not available for this example because manual intervention is required during the configuration.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To manually download and update the signature database:

1. Specify the URL for the security package.

```
[edit]
user@host#set security idp security-package url
  https://services.netscreen.com/cgi-bin/index.cgi
```

NOTE: By default it will take URL as https://services.netscreen.com/cgi-bin/index.cgi.

2. Commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

3. Switch to operational mode.

   ```
   [edit]
   user@host# exit
   ```

4. Download the security package.

   ```
   user@host>request security idp security-package download full-update
   ```

   NOTE: You can perform an offline signature package download on your device. You can download the signature package and copy the package to any common location in the device and download the package offline using the **request security idp security-package offline-download** command.

   The signature package installation remains the same and will be a full-update always.

5. Check the security package download status.

   ```
   user@host>request security idp security-package download status
   ```

6. Update the attack database using the install command.

   ```
   user@host>request security idp security-package install
   ```

7. Check the attack database update status with the following command (the command output displays information about the downloaded and installed versions of the attack database versions):

   ```
   user@host>request security idp security-package install status
   ```

8. Switch to configuration mode.

   ```
   user@host>configure
   ```

9. Create an IDP policy.

   ```
   [edit ]
   user@host#edit security idp idp-policy policy1
   ```

10. Associate attack objects or attack object groups with the policy.

   ```
   [edit security idp idp-policy policy1]
   ```

```
user@host#set rulebase-ips rule rule1 match attacks predefined-attack-groups
  "Response_Critical"
```

11. Set action.

    ```
    [edit security idp idp-policy policy1]
    user@host#set rulebase-ips rule rule1 then action no-action
    ```

12. Activate the policy.

    ```
    [edit]
    user@host#set security idp active-policy policy1
    ```

13. Commit the configuration.

    ```
    [edit]
    user@host# commit
    ```

14. After a week, download only the updates that Juniper Networks has recently uploaded.

    ```
    user@host>request security idp security-package download
    ```

15. Check the security package download status.

    ```
    user@host>request security idp security-package download status
    ```

16. Update the attack database, the active policy, and the detector with the new changes.

    ```
    user@host>request security idp security-package install
    ```

17. Check the attack database, the active policy and the detector using install status.

    ```
    user@host>request security idp security-package install status
    ```

NOTE: It is possible that an attack might be removed from the new version of an attack database. If this attack is used in an existing policy on your device, the installation of the new database will fail. An installation status message identifies the attack that is no longer valid. To update the database successfully, remove all references to the deleted attack from your existing policies and groups, and rerun the install command.

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show security idp
  idp-policy policy1 {
  rulebase-ips {
    rule rule1 {
  match {
  attacks {
    predefined-attack-groups Response_Critical;
  }
  }
    then {
  action {
  no-action;
  }
}
}
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the IDP Signature Database Manually on page 38

Verifying the IDP Signature Database Manually

Purpose Display the IDP signature database manually.

Action From operational mode, enter the **show security idp** command.

See Also

- **request security idp security-package offline-download on page 532**

Example: Downloading and Installing the IDP Security Packages in Chassis Cluster Mode

This example shows how to download and install the IDP signature database to a device operating in chassis cluster mode.

- Requirements on page 38
- Overview on page 39
- Downloading and Installing the IDP Signature Database on page 39

Requirements

Before you begin, set the chassis cluster node ID and cluster ID. See *Example: Setting the Node ID and Cluster ID for SRX Series Devices in a Chassis Cluster* .

Overview

The security package for Intrusion Detection and Prevention (IDP) contains a database of predefined IDP attack objects and IDP attack object groups that you can use in IDP policies to match traffic against known and unknown attacks. Juniper Networks regularly updates the predefined attack objects and groups with newly discovered attack patterns.

To update the signature database, you must download a security package from the Juniper Networks website. After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device.

NOTE: **On all branch SRX Series devices,, if your device memory utilization is high on the control plane, loading a large IDP policy might cause the device to run out of memory. This can trigger a system reboot during the IDP security package update.**

For more details, see "Understanding the IDP Signature Database" on page 31.

When you download the IDP security package on a device operating in chassis cluster mode, the security package is downloaded to the primary node and then synchronized to the secondary node. This synchronization helps maintain the same version of the security package on both the primary node and the secondary node.

Downloading and Installing the IDP Signature Database

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Specify the URL for the security package.

```
[edit]
user@host# set security idp security-package url
  https://services.netscreen.com/cgi-bin/index.cgi
```

2. Switch to operational mode.

```
[edit]
user@host# exit
```

3. Download the IDP security package to the primary node (downloads in the *var/db/idpd/sec-download* folder.

```
{primary:node0}[edit]
user@host> request security idp security-package download
```

The following message is displayed.

```
node0:
--------------------------------------------------------------------------
```

```
Will be processed in async mode. Check the status using the status checking
 CLI
```

4. Check the security package download status.

   ```
   {primary:node0}[edit]
   user@host> request security idp security-package download status
   ```

   On a successful download, the following message is displayed.

   ```
   node0:
   --------------------------------------------------------------------------
   Done;Successfully downloaded from
   (https://services.netscreen.com/cgi-bin/index.cgi)
   and synchronized to backup.
   Version info:1871(Mon Mar  7 09:05:30 2011, Detector=11.4.140110223)
   ```

5. Update the attack database using the **install** command.

   ```
   user@host> request security idp security-package install
   ```

6. Check the attack database update status. The command output displays information about the downloaded and installed versions of the attack database.

   ```
   {primary:node0}[edit]
   user@host> request security idp security-package install status
   ```

   ```
   node0:
   --------------------------------------------------------------------------
   Done;Attack DB update : successful - [UpdateNumber=2011,ExportDate=Mon Oct
   17 15:13:06 2011,Detector=11.6.140110920]
        Updating control-plane with new detector : successful
        Updating data-plane with new attack or detector : not performed
         due to no existing running policy found.

   node1:
   --------------------------------------------------------------------------
   Done;Attack DB update : successful - [UpdateNumber=2011,ExportDate=Mon Oct
   17 15:13:06 2011,Detector=11.6.140110920]
        Updating control-plane with new detector : successful
        Updating data-plane with new attack or detector : not performed
         due to no existing running policy found.
   ```

NOTE: You must download the IDP signature package into the primary node. This way, the security package is synchronized on the secondary node. Attempts to download the signature package to the secondary node will fail.

If you have configured a scheduled download for the security packages, the signature package files are automatically synchronized from the primary node to the backup node.

Understanding the IDP Signature Database Version

New attack objects are added to the signature database server frequently; downloading these updates and installing them on your managed devices regularly ensures that your network is effectively protected against the latest threats. As new attack objects are added to the signature database server, the version number of the database is updated with the latest database version number. Each signature database has a different version number with the latest database having the highest number.

When updating the signature database, the signature database update client connects to the Juniper Networks website and obtains the update using an HTTPS connection. This update—difference between the existing signature database and latest signature database—is calculated based on the version number that is assigned to each signature database. After you download the updates, the updated information is merged with the existing signature database and the version number is set to that of the latest signature database.

See Also

- Understanding Predefined IDP Attack Objects and Object Groups on page 92

Verifying the IDP Signature Database Version

Purpose Display the signature database version.

Action From the operational mode in the CLI, enter **show security idp security-package-version**.

Sample Output

```
user@host> show security idp security-package-version
Attack database version:31(Wed Apr 16 15:53:46 2008)
  Detector version :9.1.140080400
  Policy template version :N/A
```

Meaning The output displays the version numbers for the signature database, protocol detector, and the policy template on the IDP-enabled device. Verify the following information:

- **Attack database version**—On April 16, 2008, the version of the signature database active on the device is **31**.
- **Detector version**—Displays the version number of the IDP protocol detector currently running on the device.
- **Policy template version**—Displays the version of the policy template that is installed in the **/var/db/scripts/commit** directory when you run the **request security idp security-package install policy-templates** configuration statement in the CLI.

For a complete description of output, see the show security idp security-package-version description.

See Also

- Verifying the IDP Policy Compilation and Load Status on page 49

CHAPTER 3

Configuring IDP Policies

IDP Policies Overview

Supported Platforms SRX Series, vSRX

The Junos OS Intrusion Detection and Prevention (IDP) policy enables you to selectively enforce various attack detection and prevention techniques on network traffic passing through an IDP-enabled device. It allows you to define policy rules to match a section of traffic based on a zone, network, and application, and then take active or passive preventive actions on that traffic.

For more information, see the following topics:

- IDP Policies Overview on page 43
- Example: Enabling IDP in a Security Policy on page 45
- Verifying the IDP Policy Compilation and Load Status on page 49

IDP Policies Overview

An IDP policy defines how your device handles the network traffic. It allows you to enforce various attack detection and prevention techniques on traffic traversing your network.

A policy is made up of *rule bases*, and each rule base contains a set of *rules*. You define rule parameters, such as traffic match conditions, action, and logging requirements, then add the rules to rule bases. After you create an IDP Policy by adding rules in one or more rule bases, you can select that policy to be the active policy on your device.

Junos OS allows you to configure multiple IDP policies, but a device can have only one active IDP policy at a time. Starting with Junos OS Release 15.1X49-D20 and Junos OS Release 17.3R1, validation of configurations is done for the IDP policy that is configured as an active policy. You can install the same IDP policy on multiple devices, or you can install a unique IDP policy on each device in your network. A single policy can contain only one instance of any type of rule base.

NOTE: The IDP feature is enabled by default. No license is required. Custom attacks and custom attack groups in IDP policies can also be configured and installed even when a valid license and signature database are not installed on the device.

The following IDP policies are supported:

- DMZ_Services
- DNS_Services
- File_Server
- Getting_Started
- IDP_Default
- Recommended
- Web_Server

You can perform the following tasks to manage IDP policies:

- Create new IDP policies starting from scratch. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.
- Create an IDP policy starting with one of the predefined templates provided by Juniper Networks (see "Understanding Predefined IDP Policy Templates" on page 51).
- Add or delete rules within a rule base. You can use any of the following IDP objects to create rules:
  - Zone

    NOTE: You can configure source-address and source-except addresses when from-zone is any, and similarly to have destination-address and destination-except addresses when to-zone is any.

  - Network objects available in the base system
  - Predefined service objects provided by Juniper Networks
  - Custom application objects
  - Predefined attack objects provided by Juniper Networks
- Create custom attack objects (see "Example: Configuring IDP Signature-Based Attacks" on page 211).

- Update the signature database provided by Juniper Networks. This database contains all predefined objects.
- Maintain multiple IDP policies. Any one of the policies can be applied to the device.

The IDP policies for each user logical system are compiled together and stored on the data plane memory. To estimate adequate data plane memory for a configuration, consider these two factors:

- IDP policies applied to each user logical system are considered unique instances because the ID and zones for each user logical system are different. Estimates need to consider the combined memory requirements for all user logical systems.
- As the application database increases, compiled policies requires more memory. Memory usage should be kept below the available data plane memory to allow for database increases.

See Also

- Understanding IDP Policy Rules on page 55
- Understanding IDP Terminal Rules on page 72
- Understanding IDP Application Sets on page 218
- Understanding Custom Attack Objects on page 93

Example: Enabling IDP in a Security Policy

This example shows how to configure two security policies to enable IDP services on all HTTP and HTTPS traffic flowing in both directions on an SRX Series device. This type of configuration can be used to monitor traffic to and from a secure area of an internal network as an added security measure for confidential communications.

NOTE: In this example, Zone2 is part of the internal network.

- Requirements on page 45
- Overview on page 46
- Configuration on page 46
- Verification on page 48

Requirements

Before you begin:

- Configure network interfaces.
- Create security zones. See *Example: Creating Security Zones*.
- Configure applications. See "Example: Configuring IDP Applications and Services" on page 221.

Overview

For transit traffic to pass through IDP inspection, you configure a security policy and enable IDP application services on all traffic that you want to inspect. Security policies contain rules defining the types of traffic permitted on the network and the way that the traffic is treated inside the network. Enabling IDP in a security policy directs traffic that matches the specified criteria to be checked against the IDP rulebases.

NOTE: IDP is enabled by default. No license is required. Custom attacks and custom attack groups in IDP policies can be configured and installed even when a valid license and signature database are not installed on the device.

To allow transit traffic to pass through without IDP inspection, specify a *permit* action for the rule without enabling the IDP application services. Traffic matching the conditions in this rule passes through the device without IDP inspection.

This example shows how to configure two policies, idp-app-policy-1 and idp-app-policy-2, to enable IDP services on all HTTP and HTTPS traffic flowing in both directions on the device. The idp-app-policy-1 policy directs all HTTP and HTTPS traffic flowing from previously configured Zone1 to Zone2 to be checked against IDP rulebases. The idp-app-policy-2 policy directs all HTTP and HTTPS traffic flowing from Zone2 to Zone1 to be checked against IDP rulebases.

NOTE: The action set in the security policy action must be *permit*. You cannot enable IDP for traffic that the device denies or rejects.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security policies from-zone Zone1 to-zone Zone2 policy idp-app-policy-1 match
  source-address any
set security policies from-zone Zone1 to-zone Zone2 policy idp-app-policy-1 match
  destination-address any
set security policies from-zone Zone1 to-zone Zone2 policy idp-app-policy-1 match
  application junos-http
set security policies from-zone Zone1 to-zone Zone2 policy idp-app-policy-1 match
  application junos-https
set security policies from-zone Zone1 to-zone Zone2 policy idp-app-policy-1 then permit
  application-services idp
set security policies from-zone Zone2 to-zone Zone1 policy idp-app-policy-2 match
  source-address any
set security policies from-zone Zone2 to-zone Zone1 policy idp-app-policy-2 match
  destination-address any
```

```
set security policies from-zone Zone2 to-zone Zone1 policy idp-app-policy-2 match
   application junos-http
set security policies from-zone Zone2 to-zone Zone1 policy idp-app-policy-2 match
   application junos-https
set security policies from-zone Zone2 to-zone Zone1 policy idp-app-policy-2 then permit
   application-services idp
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable IDP services on all HTTP and HTTPS traffic flowing in both directions on the device:

1. Create a security policy for traffic flowing from Zone1 to Zone2 that has been identified as junos-http or junos-https application traffic.

   ```
   user@host# set security policies from-zone Zone1 to-zone Zone2 policy
      idp-app-policy-1 match source-address any
   user@host# set security policies from-zone Zone1 to-zone Zone2 policy
      idp-app-policy-1 match destination-address any
   user@host# set security policies from-zone Zone1 to-zone Zone2 policy
      idp-app-policy-1 match application junos-http
   user@host# set security policies from-zone Zone1 to-zone Zone2 policy
      idp-app-policy-1 match application junos-https
   ```

2. Specify the action to be taken on Zone1 to Zone2 traffic that matches conditions specified in the policy.

   ```
   user@host# set security policies from-zone Zone1 to-zone Zone2 policy
      idp-app-policy-1 then permit application-services idp
   ```

3. Create another security policy for traffic flowing in the opposite direction that has been identified as junos-http or junos-https application traffic.

   ```
   user@host# set security policies from-zone Zone2 to-zone Zone1 policy
      idp-app-policy-2 match source-address any
   user@host# set security policies from-zone Zone2 to-zone Zone1 policy
      idp-app-policy-2 match destination-address any
   user@host# set security policies from-zone Zone2 to-zone Zone1 policy
      idp-app-policy-2 match application junos-http
   user@host# set security policies from-zone Zone2 to-zone Zone1 policy
      idp-app-policy-2 match application junos-https
   ```

4. Specify the action to be taken on traffic that matches the conditions specified in this policy.

   ```
   user@host# set security policies from-zone Zone2 to-zone Zone1 policy
      idp-app-policy-2 then permit application-services idp
   ```

Results From configuration mode, confirm your configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security policies
from-zone Zone1 to-zone Zone2 {
  policy idp-app-policy-1 {
    match {
      source-address any;
      destination-address any;
      application [junos-http junos-https];
    }
    then {
      permit {
        application-services {
          idp;
        }
      }
    }
  }
}
from-zone Zone2 to-zone Zone1 {
  policy idp-app-policy-2 {
    match {
      source-address any;
      destination-address any;
      application [junos-http junos-https];
    }
    then {
      permit {
        application-services {
          idp;
        }
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 48

Verifying the Configuration

Purpose Verify that the security policy configuration is correct.

Action From operational mode, enter the **show security policies** command.

Verifying the IDP Policy Compilation and Load Status

Purpose Display the IDP log files to verify the IDP policy load and compilation status. When activating an IDP policy, you can view the IDP logs and verify if the policy is loaded and compiled successfully.

Action To track the load and compilation progress of an IDP policy, configure either one or both of the following in the CLI:

- You can configure a log file, which will be located in **/var/log/**, and set trace option flags to record these operations:

  ```
  user@host# set security idp traceoptions file idpd
  user@host# set security idp traceoptions flag all
  ```

- You can configure your device to log system log messages to a file in the **/var/log** directory:

  ```
  user@host# set system syslog file messages any any
  ```

After committing the configuration in the CLI, enter either of the following commands from the shell prompt in the UNIX-level shell:

Sample Output

```
user@host> start shell
user@host% tail -f /var/log/idpd
Aug 3 15:46:42 chiron clear-log[2655]: logfile cleared
Aug  3 15:47:12 idpd_config_read: called: check: 0
Aug  3 15:47:12 idpd commit in progres ...
Aug  3 15:47:13 Entering enable processing.
Aug  3 15:47:13 Enable value (default)
Aug  3 15:47:13 IDP processing default.
Aug  3 15:47:13 idp config knob set to (2)
Aug  3 15:47:13 Warning: active policy configured but no application package
installed, attack may not be detected!
Aug  3 15:47:13 idpd_need_policy_compile:480 Active policy path
/var/db/idpd/sets/idpengine.set
Aug  3 15:47:13 Active Policy (idpengine) rule base configuration is changed so
need to recompile active policy
Aug  3 15:47:13 Compiling policy idpengine....
Aug  3 15:47:13 Apply policy configuration, policy ops bitmask = 41
Aug  3 15:47:13 Starting policy(idpengine) compile with compress dfa...
Aug  3 15:47:35 policy compilation memory estimate: 82040
Aug  3 15:47:35 ...Passed
Aug  3 15:47:35 Starting policy package...
Aug  3 15:47:36 ...Policy Packaging Passed
Aug  3 15:47:36 [get_secupdate_cb_status] state = 0x1
Aug  3 15:47:36 idpd_policy_apply_config idpd_policy_set_config()
Aug  3 15:47:36 Reading sensor config...
Aug  3 15:47:36 sensor/idp node does not exist, apply defaults
Aug  3 15:47:36 sensor conf saved
Aug  3 15:47:36 idpd_dev_add_ipc_connection called...
Aug  3 15:47:36 idpd_dev_add_ipc_connection: done.
Aug  3 15:47:36 idpd_policy_apply_config: IDP state (2) being set
Aug  3 15:47:36 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:36 idpd_comm_server_get_event:553: evDispatch OK
```

```
Aug  3 15:47:36 Apply policy configuration, policy ops bitmask = 4
Aug  3 15:47:36 Starting policy load...
Aug  3 15:47:36 Loading policy(/var/db/idpd/bins/idpengine.bin.gz.v +
/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v +
/var/db/idpd/bins/compressed_ai.bin)...
Aug  3 15:47:36 idpd_dev_add_ipc_connection called...
Aug  3 15:47:36 idpd_dev_add_ipc_connection: done.
Aug  3 15:47:37 idpd_policy_load: creating temp tar directory
'/var/db/idpd//bins/52b58e5'
Aug  3 15:47:37 sc_policy_unpack_tgz: running addver cmd '/usr/bin/addver -r
/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v
/var/db/idpd//bins/52b58e5/__temp.tgz > /var/log/idpd.addver'
Aug  3 15:47:38 sc_policy_unpack_tgz: running tar cmd '/usr/bin/tar -C
/var/db/idpd//bins/52b58e5 -xzf /var/db/idpd//bins/52b58e5/__temp.tgz'
Aug  3 15:47:40 idpd_policy_load: running cp cmd 'cp
/var/db/idpd//bins/52b58e5/detector4.so /var/db/idpd//bins/detector.so'
Aug  3 15:47:43 idpd_policy_load: running chmod cmd 'chmod 755
/var/db/idpd//bins/detector.so'
Aug  3 15:47:44 idpd_policy_load: running rm cmd 'rm -fr
/var/db/idpd//bins/52b58e5'
Aug  3 15:47:45 idpd_policy_load: detector version: 10.3.160100209
Aug  3 15:47:45 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:45 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:45 idp_policy_loader_command: sc_klibs_subs_policy_pre_compile()
returned 0 (EOK)
Aug  3 15:47:45 idpd_policy_load: IDP_LOADER_POLICY_PRE_COMPILE returned EAGAIN,
 retrying... after (5) secs
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idp_policy_loader_command: sc_klibs_subs_policy_pre_compile()
returned 0 (EOK)
Aug  3 15:47:50 idpd_policy_load: idp policy parser pre compile succeeded, after
 (1) retries
Aug  3 15:47:50 idpd_policy_load: policy parser compile  subs s0 name
/var/db/idpd/bins/idpengine.bin.gz.v.1 buf 0x0 size 0zones 0xee34c7 z_size 136
detector /var/db/idpd//bins/detector.so ai_buf 0x0 ai_size 0 ai
/var/db/idpd/bins/compressed_ai.bin
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_policy_load: idp policy parser compile succeeded
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_policy_load: idp policy pre-install succeeded
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_policy_load: idp policy install succeeded
Aug  3 15:47:50 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:50 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:50 idpd_policy_load: idp policy post-install succeeded
Aug  3 15:47:51 IDP policy[/var/db/idpd/bins/idpengine.bin.gz.v] and
detector[/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v]
 loaded successfully.
Aug  3 15:47:51 Applying sensor configuration
Aug  3 15:47:51 idpd_dev_add_ipc_connection called...
Aug  3 15:47:51 idpd_dev_add_ipc_connection: done.
Aug  3 15:47:51 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:51 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:51 idpd_comm_server_get_event:545: evGetNext got event.
Aug  3 15:47:51 idpd_comm_server_get_event:553: evDispatch OK
Aug  3 15:47:51
...idpd commit end
Aug  3 15:47:51 Returning from commit mode, status = 0.
Aug  3 15:47:51 [get_secupdate_cb_status] state = 0x1
Aug  3 15:47:51 Got signal SIGCHLD....
```

Sample Output

```
user@host> start shell
user@host% tail -f /var/log/messages
Aug  3 15:46:56  chiron mgd[2444]: UI_COMMIT_PROGRESS: Commit operation in
progress: no commit script changes
Aug  3 15:46:56  chiron mgd[2444]: UI_COMMIT_PROGRESS: Commit operation in
progress: no transient commit script changes
Aug  3 15:46:56  chiron mgd[2444]: UI_COMMIT_PROGRESS: Commit operation in
progress: finished loading commit script changes
Aug  3 15:46:56  chiron mgd[2444]: UI_COMMIT_PROGRESS: Commit operation in
progress: exporting juniper.conf
.....
Aug  3 15:47:51  chiron idpd[2678]: IDP_POLICY_LOAD_SUCCEEDED: IDP
policy[/var/db/idpd/bins/idpengine.bin.gz.v] and
detector[/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v]
 loaded successfully(Regular load).
Aug  3 15:47:51  chiron idpd[2678]: IDP_COMMIT_COMPLETED: IDP policy commit is
complete.
......
Aug  3 15:47:51  chiron chiron sc_set_flow_max_sessions: max sessions set 16384
```

Meaning

Displays log messages showing the procedures that run in the background after you commit the **set security idp active-policy** command. This sample output shows that the policy compilation, sensor configuration, and policy load are successful.

Related Documentation

- Intrusion Detection and Prevention Overview on page 27

Predefined IDP Policy Templates

Supported Platforms SRX Series, vSRX

Juniper Networks provides predefined policy templates that you can use as a starting point for creating your own policies. Each template is set of rules of a specific rulebase type that you can copy and then update according to your requirements.

- Understanding Predefined IDP Policy Templates on page 51
- Downloading and Using Predefined IDP Policy Templates (CLI Procedure) on page 53

Understanding Predefined IDP Policy Templates

Predefined policy templates are available in the **templates.xml** file on a secured Juniper Networks website. To start using a template, you run a command from the CLI to download and copy this file to a **/var/db/scripts/commit** directory.

Each policy template contains rules that use the default actions associated with the attack objects. You should customize these templates to work on your network by selecting your own source and destination addresses and choosing IDP actions that reflect your security needs.

The client/server templates are designed for ease of use and provide balanced performance and coverage. The client/server templates include client protection, server protection, and client/server protection.

Each of the client/server templates has two versions that are device specific, a 1-gigabyte (GB) version and a 2-GB version.

NOTE: The 1-gigabyte versions labeled *1G* should only be used for devices that are limited to 1 GB of memory. If a 1-GB device loads anything other than a 1-GB policy, the device might experience policy compilation errors due to limited memory or limited coverage. If a 2-GB device loads anything other than a 2-GB policy, the device might experience limited coverage.

Use these templates as a guideline for creating policies. We recommend that you make a copy of these templates and use the copy (not the original) for the policy. This approach allows you to make changes to the policy and to avoid future issues due to changes in the policy templates.

Table 3 on page 52 summarizes the predefined IDP policy templates provided by Juniper Networks.

Table 3: Predefined IDP Policy Templates

Template Name	Description
Client-And-Server-Protection	Designed to protect both clients and servers. To be used on high memory devices with 2 GB or more of memory.
Client-And-Server-Protection-1G	Designed to protect both clients and servers. To be used on all devices, including low-memory branch devices.
Client-Protection	Designed to protect clients. To be used on high memory devices with 2 GB or more of memory.
Client-Protection-1G	Designed to protect clients. To be used on all devices, including low-memory branch devices.
DMZ Services	Protects a typical demilitarized zone (DMZ) environment.
DNS Server	Protects Domain Name System (DNS) services.
File Server	Protects file sharing services, such as Network File System (NFS), FTP, and others.
Getting Started	Contains very open rules. Useful in controlled lab environments, but should not be deployed on heavy traffic live networks.
IDP Default	Contains a good blend of security and performance.

Table 3: Predefined IDP Policy Templates (continued)

Template Name	Description
Recommended	Contains only the attack objects tagged as *recommended* by Juniper Networks. All rules have their Actions column set to take the recommended action for each attack object.
Server-Protection	Designed to protect servers. To be used on high memory devices with 2 GB or more of memory.
Server-Protection-1G	Designed to protect servers. To be used on all devices, including low-memory branch devices.
Web Server	Protects HTTP servers from remote attacks.

To use predefined policy templates:

1. Download the policy templates from the Juniper Networks website.
2. Install the policy templates.
3. Enable the **templates.xml** script file. Commit scripts in the **/var/db/scripts/commit** directory are ignored if they are not enabled.
4. Choose a policy template that is appropriate for you and customize it if you need to.
5. Activate the policy that you want to run on the system. Activating the policy might take a few minutes. Even after a commit complete message is displayed in the CLI, the system might continue to compile and push the policy to the data plane.

   NOTE: Occasionally, the compilation process might fail for a policy. In this case, the active policy showing in your configuration might not match the actual policy running on your device. Run the **show security idp status** command to verify the running policy. Additionally, you can view the IDP log files to verify the policy load and compilation status.

6. Delete or deactivate the commit script file. By deleting the commit script file, you avoid the risk of overwriting modifications to the template when you commit the configuration. Deactivating the statement adds an inactive tag to the statement, effectively commenting out the statement from the configuration. Statements marked inactive do not take effect when you issue the **commit** command.

Downloading and Using Predefined IDP Policy Templates (CLI Procedure)

Before you begin, configure network interfaces. See the *Junos OS Interfaces Configuration Guide for Security Devices*.

To download and use a predefined policy template:

1. Download the script file **templates.xml** to the**/var/db/idpd/sec-download/sub-download** directory. This script file contains predefined IDP policy templates.

   ```
   user@host> request security idp security-package download policy-templates
   ```

2. Copy the **templates.xml** file to the **/var/db/scripts/commit** directory and rename it to **templates.xsl**.

   ```
   user@host> request security idp security-package install policy-templates
   ```

3. Enable the **templates.xsl** scripts file. At commit time, the Junos OS management process (mgd) looks in the **/var/db/scripts/commit** directory for scripts and runs the script against the candidate configuration database to ensure the configuration conforms to the rules dictated by the scripts.

   ```
   user@host# set system scripts commit file templates.xsl
   ```

4. Commit the configuration. Committing the configuration saves the downloaded templates to the Junos OS configuration database and makes them available in the CLI at the **[edit security idp idp-policy]** hierarchy level.

5. Display the list of downloaded templates.

   ```
   user@host#set security idp active-policy ?

   Possible completions:
    <active policy> Set active policy
     DMZ_Services
     DNS_Service
     File_Server
     Getting_Started
     IDP_Default
     Recommended
     Web_Server
   ```

6. Activate the predefined policy. The following statement specifies the *Recommended* predefined IDP policy as the active policy:

   ```
   user@host# set security idp active-policy Recommended
   ```

7. Delete or deactivate the commit script file. By deleting the commit script file, you avoid the risk of overwriting modifications to the template when you commit the configuration. Run one of the following commands:

   ```
   user@host# delete system scripts commit file templates.xsl
   user@host# deactivate system scripts commit file templates.xsl
   ```

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

9. You can verify the configuration by using the **show security idp status** command. For more information, see the *Junos OS CLI Reference*.

Related Documentation

- IDP Application Identification on page 225

IDP Policy Rules and IDP Rule Bases

Supported Platforms SRX Series, vSRX

Intrusion Detection and Prevention (IDP) policies are collections of rules and rulebases.

For more information, see the following topics:

Understanding IDP Policy Rule Bases

A rulebase is an ordered set of rules that use a specific detection method to identify and prevent attacks.

Rules are instructions that provide context to detection mechanisms by specifying which part of the network traffic the IDP system should look in to find attacks. When a rule is matched, it means that an attack has been detected in the network traffic, triggering the action for that rule. The IDP system performs the specified action and protects your network from that attack.

Each rulebase can have multiple rules—you determine the sequence in which rules are applied to network traffic by placing them in the desired order. Each rulebase in the IDP system uses specific detection methods to identify and prevent attacks. Junos OS supports two types of rulebases—intrusion prevention system (IPS) rulebase and exempt rulebase.

Understanding IDP Policy Rules

Each instruction in an Intrusion Detection and Prevention (IDP) policy is called a rule. Rules are created in rulebases.

Rulebases are a set of rules that combine to define an IDP policy. Rules provide context to detection mechanisms by specifying which part of the network traffic the IDP system should look in to find attacks. When a rule is matched, it means that an attack has been detected in the network traffic, triggering the action for that rule. The IDP system performs the specified action and protects your network from that attack.

IDP policy rules are made up of the following components:

- Understanding IDP Rule Match Conditions on page 56
- Understanding IDP Rule Objects on page 56
- Understanding IDP Rule Actions on page 59
- Understanding IDP Rule IP Actions on page 61
- Understanding IDP Rule Notifications on page 62

Understanding IDP Rule Match Conditions

Match conditions specify the type of network traffic you want IDP to monitor for attacks.

Match conditions use the following characteristics to specify the type of network traffic to be monitored:

- **From-zone** and **to-zone**—All traffic flows from a source to a destination zone. You can select any zone for the source or destination. You can also use zone exceptions to specify unique to and from zones for each device. Specify **any** to monitor network traffic originating from and to any zone. The default value is **any**.

  NOTE: You can now specify **source-address** and **source-except** addresses when **from-zone** is **any**. Similarly, when **to-zone** is **any**, you can specify **destination-address** and **destination-except** addresses.

- **Source IP address**—Specify the source IP address from which the network traffic originates. You can specify **any** to monitor network traffic originating from any IP address. You can also specify **source-except** to specify all sources except the specified addresses. The default value is **any**.
- **Destination IP address**—Specify the destination IP address to which the network traffic is sent. You can set this to **any** to monitor network traffic sent to any IP address. You can also specify **destination-except** to specify all destinations except the specified addresses. The default value is **any**.
- **Application**—Specify the Application Layer protocols supported by the destination IP address. You can specify **any** for all applications and **default** for the application configured in the attack object for the rule.

Understanding IDP Rule Objects

Objects are reusable logical entities that you can apply to rules. Each object that you create is added to a database for the object type.

You can configure the following types of objects for IDP rules.

Zone Objects

A zone or security zone is a collection of one or more network interfaces. IDP uses zone objects configured in the base system.

Address or Network Objects

Address objects represent components of your network, such as host machines, servers, and subnets. You use address objects in IDP policy rules to specify the network components that you want to protect.

Application or Service Objects

Service objects represent network services that use Transport Layer protocols such as TCP, UDP, RPC, and ICMP. You use service objects in rules to specify the service an attack uses to access your network. Juniper Networks provides predefined service objects, a database of service objects that are based on industry-standard services. If you need to add service objects that are not included in the predefined service objects, you can create custom service objects. IDP supports the following types of service objects:

- **Any**—Allows IDP to match all Transport Layer protocols.
- **TCP**—Specifies a TCP port or a port range to match network services for specified TCP ports. You can specify **junos-tcp-any** to match services for all TCP ports.
- **UDP**—Specifies a UDP port or a port range to match network services for specified UDP ports. You can specify **junos-udp-any** to match services for all UDP ports.
- **RPC**—Specifies a remote procedure call (RPC from Sun Microsystems) program number or a program number range. IDP uses this information to identify RPC sessions.
- **ICMP**—Specifies a type and code that is a part of an ICMP packet. You can specify **junos-icmp-all** to match all ICMP services.
- **default**—Allows IDP to match default and automatically detected protocols to the applications implied in the attack objects.

Attack Objects

IDP attack objects represent known and unknown attacks. IDP includes a predefined attack object database that is periodically updated by Juniper Networks. Attack objects are specified in rules to identify malicious activity. Each attack is defined as an attack object, which represents a known pattern of attack. Whenever this known pattern of attack is encountered in the monitored network traffic, the attack object is matched. The three main types of attack objects are described in Table 4 on page 58:

Table 4: IDP Attack Objects Description

Attack Objects	Description
Signature Attack Objects	Signature attack objects detect known attacks using stateful attack signatures. An attack signature is a pattern that always exists within an attack; if the attack is present, so is the attack signature. With stateful signatures, IDP can look for the specific protocol or service used to perpetrate the attack, the direction and flow of the attack, and the context in which the attack occurs. Stateful signatures produce few false positives because the context of the attack is defined, eliminating huge sections of network traffic in which the attack would not occur.
Protocol Anomaly Attack Objects	Protocol anomaly attack objects identify unusual activity on the network. They detect abnormal or ambiguous messages within a connection according to the set of rules for the particular protocol being used. Protocol anomaly detection works by finding deviations from protocol standards, most often defined by RFCs and common RFC extensions. Most legitimate traffic adheres to established protocols. Traffic that does not, produces an anomaly, which may be created by attackers for specific purposes, such as evading an intrusion prevention system (IPS).
Compound Attack Objects	A compound attack object combines multiple signatures and/or protocol anomalies into a single object. Traffic must match all of the combined signatures and/or protocol anomalies to match the compound attack object; you can specify the order in which signatures or anomalies must match. Use compound attack objects to refine your IDP policy rules, reduce false positives, and increase detection accuracy. A compound attack object enables you to be very specific about the events that need to occur before IDP identifies traffic as an attack. You can use **And**, **Or**, and **Ordered and** operations to define the relationship among different attack objects within a compound attack and the order in which events occur.

Attack Object Groups

IDP contains a large number of predefined attack objects. To help keep IDP policies organized and manageable, attack objects can be grouped. An attack object group can contain one or more attack objects of different types. Junos OS supports the following three types of attack groups:

- Predefined attack object groups—Contain objects present in the signature database. The predefined attack object groups are dynamic in nature. For example, FTP: Minor group selects all attacks of application- FTP and severity- Minor. If a new FTP attack of minor severity is introduced in the security database, it is added to the FTP: Minor group by default.
- Dynamic attack groups—Contain attack objects based on a certain matching criteria. For example, a dynamic group can contain all attacks related to an application. During signature update, the dynamic group membership is automatically updated based on the matching criteria for that group.

  On SRX Series devices, for a dynamic attack group using the direction filter, the expression **and** should be used in the exclude values. As is the case with all filters, the default expression is **or**. However, there is a choice of **and** in the case of the direction filter.

For example, if you want to choose all attacks with the direction client-to-server, configure the direction filter using **set security idp dynamic-attack-group dyn1 filters direction values client-to-server** command

In the case of chain attacks, each of the multiple members has its own direction. If a policy includes chain attacks, a client-to-server filter selects all chain attacks that have any member with client-to-server as the direction. This means chain attacks that include members with server-to-client or ANY as the direction are selected if the chain has at least one member with client-to-server as the direction.

To prevent these chain attacks from being added to the policy, configure the dynamic group as follows:

- **set security idp dynamic-attack-group dyn1 filters direction expression and**
- **set security idp dynamic-attack-group dyn1 filters direction values client-to-server**
- **set security idp dynamic-attack-group dyn1 filters direction values exclude-server-to-client**
- **set security idp dynamic-attack-group dyn1 filters direction values exclude-any**

- Custom attack groups—Contain customer-defined attack groups and can be configured through the CLI. They can contain specific predefined attacks, custom attacks, predefined attack groups, or dynamic attack groups. They are static in nature, because the attacks are specified in the group. Therefore the attack groups do not change when the security database is updated

Understanding IDP Rule Actions

Actions specify the actions you want IDP to take when the monitored traffic matches the attack objects specified in the rules.

Table 5 on page 59 shows the actions you can specify for IDP rules:

Table 5: IDP Rule Actions

Term	Definition
No Action	No action is taken. Use this action when you only want to generate logs for some traffic.
Ignore Connection	Stops scanning traffic for the rest of the connection if an attack match is found. IDP disables the rulebase for the specific connection. NOTE: This action does not mean ignore an attack.
Diffserv Marking	Assigns the indicated Differentiated Services code point (DSCP) value to the packet in an attack, then passes the packet on normally. Note that DSCP value is not applied to the first packet that is detected as an attack, but is applied to subsequent packets.

Table 5: IDP Rule Actions (continued)

Term	Definition
Drop Packet	Drops a matching packet before it can reach its destination but does not close the connection. Use this action to drop packets for attacks in traffic that is prone to spoofing, such as UDP traffic. Dropping a connection for such traffic could result in a denial of service that prevents you from receiving traffic from a legitimate source-IP address. NOTE: When an IDP policy is configured using a non-packet context defined in a custom signature for any application and has the action drop packet, when IDP identifies an attack the decoder will promote drop_packet to drop_connection. With a DNS protocol attack, this is not the case. The DNS decoder will not promote drop_packet to drop_connection when an attack is identified. This will ensure that only DNS attack traffic will be dropped and valid DNS requests will continue to be processed. This will also avoid TCP retransmission for the valid TCP DNS requests.
Drop Connection	Drops all packets associated with the connection, preventing traffic for the connection from reaching its destination. Use this action to drop connections for traffic that is not prone to spoofing.
Close Client	Closes the connection and sends an RST packet to the client but not to the server.
Close Server	Closes the connection and sends an RST packet to the server but not to the client.
Close Client and Server	Closes the connection and sends an RST packet to both the client and the server.
Recommended	All predefined attack objects have a default action associated with them. This is the action that Juniper Networks recommends when that attack is detected. NOTE: This action is supported only for IPS rulebases. Recommended —A list of all attack objects that Juniper Networks considers to be serious threats, organized into categories. • Attack type groups attack objects by type (anomaly or signature). Within each type, attack objects are grouped by severity. • Category groups attack objects by predefined categories. Within each category, attack objects are grouped by severity. • Operating system groups attack objects by the operating system to which they apply: BSD, Linux, Solaris, or Windows. Within each operating system, attack objects are grouped by services and severity. • Severity groups attack objects by the severity assigned to the attack. IDP has five severity levels: Critical, Major, Minor, Warning, and Info. Within each severity, attack objects are grouped by category.

Understanding IDP Rule IP Actions

IP actions are actions that apply on future connections that use the same IP action attributes. For example, you can configure an IP action in the rule to block all future HTTP sessions between two hosts if an attack is detected on a session between the hosts. Or you can specify a timeout value that defines that the action should be applied only if new sessions are initiated within that specified timeout value. The default timeout value for IP actions is 0, which means that IP actions are never timed out.

IP actions are similar to other actions; they direct IDP to drop or close the connection. However, because you now also have the attacker's IP address, you can choose to block the attacker for a specified time. If attackers cannot immediately regain a connection to your network, they might try to attack easier targets. Use IP actions in conjunction with actions and logging to secure your network.

IP action attributes are a combination of the following fields:

- Source IP address
- Destination IP address
- Destination port
- From-zone
- Protocol

Table 6 on page 61 summarizes the types IP actions supported by IDP rules:

Table 6: IDP Rule IP Actions

Term	Definition
Notify	Does not take any action against future traffic, but logs the event. This is the default.
Drop/Block Session	All packets of any session matching the IP action rule are dropped silently.
Close Session	Any new sessions matching this IP action rule are closed by sending RST packets to the client and server.

When traffic matches multiple rules, the most severe IP action of all matched rules is applied. The most severe IP action is the Close Session action, the next in severity is the Drop/Block Session action, and then the Notify action.

NOTE: After enhancements to the central point, the system has the following limitations:

- The maximum active mode **ip-action** number for each SPU is limited to 600000 entries. When this limit is reached, you cannot create a new active mode **ip-action** entry on the SPU.
- The maximum all modes (active mode and passive mode) **ip-action** number for each SPU is limited to 1200000 entries. When this limit is reached, you cannot create a new active mode **ip-action** entry on the SPU.
- When you run the **clear ip-action** command, the **ip-action** entries are deleted through ring messages. When the CPU usage is high, the deleted ring messages are dropped and resent by the active mode **ip-action**. As the deleting process takes time, you can see few **ip-action** entries when you run the **show ip-action** command.

On devices where central point enhancements are not done, only active mode **ip-action** exists and the maximum **ip-action** number is limited to 600000. When this limit is reached, you cannot create a new active mode **ip-action** entry.

Understanding IDP Rule Notifications

Notification defines how information is to be logged when an action is performed. When attacks are detected, you can choose to log an attack and create log records with attack information and send that information to the log server.

By using notifications, you can also configure the following options that instruct the log server to perform specific actions on logs generated for each rule:

- **Set Alerts**—Specify an alert option for a rule in the IDP policy. When the rule is matched, the corresponding log record displays an alert in the alert column of the Log Viewer. Security administrators use alerts to become aware of and react to important security events.
- **Set Severity Level**—Set severity levels in logging to support better organization and presentation of log records on the log server. You can use the default severity settings of the selected attack objects or choose a specific severity for your rule. The severity you configure in the rules overrides the inherited attack severity. You can set the severity level to the following levels:
  - Info—2
  - Warning—3
  - Minor—4
  - Major—5
  - Critical—7

Example: Inserting a Rule in the IDP Rulebase

This example shows how to insert a rule in the IDP rulebase.

Requirements

Before you begin:

- Configure network interfaces.
- Define rules in a rulebase. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.

Overview

The IDP rule-matching algorithm starts from the top of the rulebase and checks traffic against all rules in the rulebase that match the specified match conditions. You determine the sequence in which rules are applied to network traffic by placing them in the desired order. When you add a rule to the rulebase, it is placed at the end of the existing list of rules. To place a rule in any other location than at the end of the rulebase, you *insert* the rule at the desired location in the rulebase. This example places rule R2 before rule R1 in the IDP IPS rulebase in a policy called base-policy.

Configuration

Step-by-Step Procedure

To insert a rule in the rulebase:

1. Define the position of the rule in the rulebase based on the order in which you want the rule to be evaluated.

```
[edit]
user@host# insert security idp idp-policy base-policy rulebase-ips rule R2 before
  rule R1
```

2. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

Verification

To verify the configuration is working properly, enter the **show security idp status** command.

Example: Deactivating and Activating Rules in an IDP Rulebase

This example shows how to deactivate and activate a rule in a rulebase.

Requirements

Before you begin:

- Configure network interfaces.

- Define rules in a rulebase. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.

Overview

In a rulebase, you can disable and enable rules by using the **deactivate** and **activate** commands. The **deactivate** command comments out the specified statement from the configuration. Rules that have been deactivated do not take effect when you issue the **commit** command. The **activate** command adds the specified statement back to the configuration. Rules that have been activated take effect when you next issue the **commit** command. This example shows how to deactivate and reactivate rule R2 in an IDP IPS rulebase that is associated with a policy called base-policy.

Configuration

Step-by-Step Procedure

To deactivate and activate a rule in a rulebase:

1. Specify the rule that you want to deactivate.

   ```
   [edit]
   user@host# deactivate security idp idp-policy base-policy rulebase-ips rule R2
   ```

2. Activate the rule.

   ```
   [edit]
   user@host# activate security idp idp-policy base-policy rulebase-ips rule R2
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

Verification

To verify the configuration is working properly, enter the **show security idp status** command.

Understanding IDP Application-Level DDoS Rulebases

The application-level DDoS rulebase defines parameters used to protect servers, such as DNS or HTTP, from application-level distributed denial-of-service (DDoS) attacks. You can set up custom application metrics based on normal server activity requests to determine when clients should be considered an attack client. The application-level DDoS rulebase is then used to defines the source match condition for traffic that should be monitored, then takes the defined action: close server, drop connection, drop packet, or no action. It can also perform an IP action: ip-block, ip-close, ip-notify, ip-connection-rate-limit, or timeout. Table 7 on page 65 summarizes the options that you can configure in the application-level DDoS rulebase rules.

Table 7: Application-Level DDoS Rulebase Components

Term	Definition
Match condition	Specify the network traffic you want the device to monitor for attacks.
Action	Specify the actions you want Intrusion Detection and Prevention (IDP) to take when the monitored traffic matches the application-ddos objects specified in the application-level DDoS rule.
IP Action	Enables you to implicitly block a source address to protect the network from future intrusions while permitting legitimate traffic. You can configure one of the following IP action options in application-level DDoS: ip-block, ip-close, ip-notify, and ip-connection-rate-limit.

Understanding IDP IPS Rulebases

The intrusion prevention system (IPS) rulebase protects your network from attacks by using attack objects to detect known and unknown attacks. It detects attacks based on stateful signature and protocol anomalies. Table 8 on page 65 summarizes the options that you can configure in the IPS-rulebase rules.

Table 8: IPS Rulebase Components

Term	Definition
Match condition	Specify the type of network traffic you want the device to monitor for attacks. For more information about match conditions, see "Understanding IDP Policy Rules" on page 55.
Attack objects/groups	Specify the attacks you want the device to match in the monitored network traffic. Each attack is defined as an attack object, which represents a known pattern of attack. For more information about attack objects, see "Understanding IDP Policy Rules" on page 55.
Terminal flag	Specify a terminal rule. The device stops matching rules for a session when a terminal rule is matched. For more information about terminal rules, see "Understanding IDP Terminal Rules" on page 72.
Action	Specify the action you want the system to take when the monitored traffic matches the attack objects specified in the rules. If an attack triggers multiple rule actions, then the most severe action among those rules is executed. For more information about actions, see "Understanding IDP Policy Rules" on page 55.
IP Action	Enables you to protect the network from future intrusions while permitting legitimate traffic. You can configure one of the following IP action options in the IPS rulebase—notify, drop, or close. For more information about IP actions, see "Understanding IDP Policy Rules" on page 55.
Notification	Defines how information is to be logged when action is performed. You can choose to log an attack, create log records with the attack information, and send information to the log server. For more information, see "Understanding IDP Policy Rules" on page 55.

Example: Defining Rules for an IDP IPS RuleBase

This example shows how to define rules for an IDP IPS rulebase.

Requirements

Before you begin:

- Configure network interfaces.
- Create security zones. See *Example: Creating Security Zones*.
- Enable IDP in security policies. See "Example: Enabling IDP in a Security Policy" on page 45.

NOTE: For using IDP custom policy with pre-defined attacks, you need to have Signature database downloaded on the device.

For more details see "Example: Updating the IDP Signature Database Manually" on page 35.

Overview

Each rule is composed of match conditions, objects, actions, and notifications. When you define an IDP rule, you must specify the type of network traffic you want IDP to monitor for attacks by using the following characteristics—source zone, destination zone, source IP address, destination IP address, and the Application Layer protocol supported by the destination IP address. The rules are defined in rulebases, and rulebases are associated with policies.

This example describes how to create a policy called base-policy, specify a rulebase for this policy, and then add rule R1 to this rulebase. In this example, rule R1:

- Specifies the match condition to include any traffic from a previously configured zone called *trust* to another previously configured zone called *untrust*. The match condition also includes a predefined attack group Critical - TELNET. The application setting in the match condition is *default* and matches any application configured in the attack object.
- Specifies an action to drop connection for any traffic that matches the criteria for rule R1.
- Enables attack logging and specifies that an alert flag is added to the attack log.
- Specifies a severity level as *critical*.

After defining the rule, you specify base-policy as the active policy on the device.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy base-policy
set security idp idp-policy base-policy rulebase-ips rule R1 match from-zone trust to-zone
   untrust source-address any destination-address any application default
set security idp idp-policy base-policy rulebase-ips rule R1 match attacks
   predefined-attack-groups "TELNET-Critical"
set security idp idp-policy base-policy rulebase-ips rule R1 then action drop-connection
set security idp idp-policy base-policy rulebase-ips rule R1 then notification log-attacks
   alert
set security idp idp-policy base-policy rulebase-ips rule R1 then severity critical
set security idp active-policy base-policy
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To define rules for an IDP IPS rulebase:

1. Create a policy by assigning a meaningful name to it.

   ```
   [edit]
   user@host# edit security idp idp-policy base-policy
   ```

2. Associate a rulebase with the policy.

   ```
   [edit security idp idp-policy base-policy]
   user@host# edit rulebase-ips
   ```

3. Add rules to the rulebase.

   ```
   [edit security idp idp-policy base-policy rulebase-ips]
   user@host# edit rule R1
   ```

4. Define the match criteria for the rule.

   ```
   [edit security idp idp-policy base-policy rulebase-ips rule R1]
   user@host# set match from-zone trust to-zone untrust source-address any
      destination-address any application default
   ```

5. Define an attack as match criteria.

   ```
   [edit security idp idp-policy base-policy rulebase-ips rule R1]
   user@host# set match attacks predefined-attack-groups "TELNET-Critical"
   ```

6. Specify an action for the rule.

```
[edit security idp idp-policy base-policy rulebase-ips rule R1]
user@host# set then action drop-connection
```

7. Specify notification and logging options for the rule.

```
[edit security idp idp-policy base-policy rulebase-ips rule R1]
user@host# set then notification log-attacks alert
```

8. Set the severity level for the rule.

```
[edit security idp idp-policy base-policy rulebase-ips rule R1]
user@host# set then severity critical
```

9. Activate the policy.

```
[edit]
user@host# set security idp active-policy base-policy
```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy base-policy {
  rulebase-ips {
    rule R1 {
      match {
        from-zone trust;
        source-address any;
        to-zone untrust;
        destination-address any;
        application default;
        attacks {
          predefined-attack-groups Critical-TELNET;
        }
      }
      then {
        action {
          drop-connection;
        }
        notification {
          log-attacks {
            alert;
          }
        }
        severity critical;
      }
    }
  }
}
active-policy base-policy;
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 69

Verifying the Configuration

Purpose Verify that the rules for the IDP IPS rulebase configuration are correct.

Action From operational mode, enter the **show security idp status** command.

Understanding IDP Exempt Rulebases

The exempt rulebase works in conjunction with the intrusion prevention system (IPS) rulebase to prevent unnecessary alarms from being generated. You configure rules in this rulebase to exclude known false positives or to exclude a specific source, destination, or source/destination pair from matching an IPS rule. If traffic matches a rule in the IPS rulebase, the system attempts to match the traffic against the exempt rulebase before performing the action specified. Carefully written rules in an exempt rulebase can significantly reduce the number of false positives generated by an IPS rulebase.

Configure an exempt rulebase in the following conditions:

- When an IDP rule uses an attack object group that contains one or more attack objects that produce false positives or irrelevant log records.
- When you want to exclude a specific source, destination, or source/destination pair from matching an IDP rule. This prevents IDP from generating unnecessary alarms.

NOTE: **Make sure to configure the IPS rulebase before configuring the exempt rulebase.**

Table 9 on page 69 summarizes the options that you can configure in the exempt-rulebase rules.

Table 9: Exempt Rulebase Options

Term	Definition
Match condition	Specify the type of network traffic you want the device to monitor for attacks in the same way as in the IPS rulebase. However, in the exempt rulebase, you cannot configure an application; it is always set to **any**.
Attack objects/groups	Specify the attack objects that you do *not* want the device to match in the monitored network traffic.

Example: Defining Rules for an IDP Exempt Rulebase

This example shows how to define rules for an exempt IDP rulebase.

- Requirements on page 70
- Overview on page 70
- Configuration on page 70
- Verification on page 72

Requirements

Before you begin, create rules in the IDP IPS rulebase. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.

Overview

When you create an exempt rule, you must specify the following:

- Source and destination for traffic you want to exempt. You can set the source or destination to **Any** to exempt network traffic originating from any source or sent to any destination. You can also set **source-except** or **destination-except** to specify all the sources or destinations except the specified source or destination addresses.

  NOTE: **You can now specify source-address and source-except addresses when from-zone is any. Similarly, when to-zone is any, you can specify destination-address and destination-except addresses.**

- The attacks you want IDP to exempt for the specified source/destination addresses. You must include at least one attack object in an exempt rule.

This example shows that the IDP policy generates false positives for the attack FTP:USER:ROOT on an internal network. You configure the rule to exempt attack detection for this attack when the source IP is from your internal network.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy base-policy
set security idp idp-policy base-policy rulebase-exempt rule R1 match from-zone trust
  to-zone any
set security idp idp-policy base-policy rulebase-exempt rule R1 match source-address
  internal-devices destination-address any
set security idp idp-policy base-policy rulebase-exempt rule R1 match attacks
  predefined-attacks "FTP:USER:ROOT"
set security idp active-policy base-policy
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To define rules for an exempt IDP rulebase:

1. Specify the IDP IPS rulebase for which you want to define and exempt the rulebase.

   ```
   [edit]
   user@host# edit security idp idp-policy base-policy
   ```

2. Associate the exempt rulebase with the policy and zones, and add a rule to the rulebase.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-exempt rule R1 match from-zone trust to-zone any
   ```

3. Specify the source and destination addresses for the rulebase.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-exempt rule R1 match source-address internal-devices
     destination-address any
   ```

4. Specify the attacks that you want to exempt from attack detection.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-exempt rule R1 match attacks predefined-attacks
     "FTP:USER:ROOT"
   ```

5. Activate the policy.

   ```
   [edit]
   user@host# set security idp active-policy base-policy
   ```

Results

From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy base-policy {
  rulebase-exempt {
    rule R1 {
      match {
        from-zone trust;
        source-address internal-devices;
        to-zone any;
        destination-address any;
        attacks {
          predefined-attacks FTP:USER:ROOT;
        }
      }
    }
```

```
  }
active-policy base-policy;
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 72

Verifying the Configuration

Purpose Verify that the defined rules were exempt from the IDP rulebase configuration.

Action From operational mode, enter the **show security idp status** command.

Understanding IDP Terminal Rules

The Intrusion Detection and Prevention (IDP) rule-matching algorithm starts from the top of the rulebase and checks traffic against all rules in the rulebase that match the source, destination, and service. However, you can configure a rule to be *terminal*. A terminal rule is an exception to this algorithm. When a match is discovered in a terminal rule for the source, destination, zones, and application, IDP does not continue to check subsequent rules for the same source, destination, and application. It does not matter whether or not the traffic matches the attack objects in the matching rule.

You can use a terminal rule for the following purposes:

- To set different actions for different attacks for the same Source and Destination.
- To disregard traffic that originates from a known trusted source. Typically, the action is **None** for this type of terminal rule.
- To disregard traffic sent to a server that is vulnerable only to a specific set of attacks. Typically, the action is **Drop Connection** for this type of terminal rule.

Use caution when defining terminal rules. An inappropriate terminal rule can leave your network open to attacks. Remember that traffic matching the source, destination, and application of a terminal rule is not compared to subsequent rules, even if the traffic does not match an attack object in the terminal rule. Use a terminal rule only when you want to examine a certain type of traffic for one specific set of attack objects. Be particularly careful about terminal rules that use **any** for both the source and destination. Terminal rules should appear near the top of the rulebase before other rules that would match the same traffic.

Example: Setting Terminal Rules in Rulebases

This example shows how to configure terminal rules.

- Requirements on page 73
- Overview on page 73

- Configuration on page 73
- Verification on page 75

Requirements

Before you begin:

- Configure network interfaces.
- Enable IDP application services in a security policy. See "Example: Enabling IDP in a Security Policy" on page 45.
- Create security zones. See *Example: Creating Security Zones*.
- Define rules. See "Example: Inserting a Rule in the IDP Rulebase" on page 63.

Overview

By default, rules in the IDP rulebase are not terminal, which means IDP examines all rules in the rulebase and executes all matches. You can specify that a rule is terminal; that is, if IDP encounters a match for the source, destination, and service specified in a terminal rule, it does not examine any subsequent rules for that connection.

This example shows how to configure terminal rules. You define rule R2 to terminate the match algorithm if the source IP of the traffic originates from a known trusted network in your company. If this rule is matched, IDP disregards traffic from the trusted network and does not monitor the session for malicious data.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy base-policy
set security idp idp-policy base-policy rulebase-ips rule R2 match source-address internal
  destination-address any
set security idp idp-policy base-policy rulebase-ips rule R2 terminal
set security idp idp-policy base-policy rulebase-ips rule R2 match attacks
  predefined-attacks FTP:USER:ROOT
set security idp idp-policy base-policy rulebase-ips rule R2 then action recommended
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see*Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure terminal rules:

1. Create an IDP policy.

   ```
   [edit]
   user@host# set security idp idp-policy base-policy
   ```

2. Define a rule and set its match criteria.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-ips rule R2 match source-address internal
     destination-address any
   ```

3. Set the terminal flag for the rule.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-ips rule R2 terminal
   ```

4. Specify the attacks that you want to exempt from attack detection.

   ```
   [edit security idp idp-policy base-policy]
   user@host# set rulebase-ips rule R2 match attacks predefined-attacks
     FTP:USER:ROOT
   ```

5. Specify an action for the rule.

   ```
   [edit security idp idp-policy base-policy]
   user@host# rulebase-ips rule R2 then action recommended
   ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
  idp-policy base-policy {
    rulebase-ips {
      rule R2 {
        match {
          source-address internal;
          destination-address any;
          attacks {
            predefined-attacks FTP:USER:ROOT;
          }
        }
        then {
          action {
            recommended;
          }
        }
        terminal;
      }
    }
  }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 75

Verifying the Configuration

Purpose Verify that the terminal rules were configured correctly.

Action From operational mode, enter the **show security idp status** command.

Understanding DSCP Rules in IDP Policies

Differentiated Services code point (DSCP) is an integer value encoded in the 6-bit field defined in IP packet headers. It is used to enforce class-of-service (CoS) distinctions. CoS allows you to override the default packet forwarding behavior and assign service levels to specific traffic flows.

You can configure DSCP value as an action in an IDP policy rule. You first define the traffic by defining match conditions in the IDP policy and then associate a DiffServ marking action with it. Based on the DSCP value, behavior aggregate classifiers set the forwarding class and loss priority for the traffic deciding the forwarding treatment the traffic receives.

All packets that match the IDP policy rule have the CoS field in their IP header rewritten with the DSCP value specified in the matching policy. If the traffic matches multiple rules with differing DSCP values, the first IDP rule that matches takes effect and this IDP rule then applies to all traffic for that session.

Example: Configuring DSCP Rules in an IDP Policy

This example shows how to configure DSCP values in an IDP policy.

- Requirements on page 75
- Overview on page 76
- Configuration on page 76
- Verification on page 78

Requirements

Before you begin:

- Configure network interfaces.
- Enable IDP application services in a security policy. See “Example: Enabling IDP in a Security Policy” on page 45.
- Create security zones. See *Example: Creating Security Zones*.
- Define rules. See “Example: Inserting a Rule in the IDP Rulebase” on page 63.

Overview

Configuring DSCP values in IDP policies provides a method of associating CoS values—thus different levels of reliability—for different types of traffic on the network.

This example shows how to create a policy called policy1, specify a rulebase for this policy, and then add rule R1 to this rulebase. In this example, rule R1:

- Specifies the match condition to include any traffic from a previously configured zone called trust to another previously configured zone called untrust. The match condition also includes a predefined attack group called HTTP - Critical. The application setting in the match condition is specified as the default and matches any application configured in the attack object.
- Specifies an action to rewrite the CoS field in the IP header with the DSCP value 50 for any traffic that matches the criteria for rule R1.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy base-policy
set security idp idp-policy base-policy rulebase-ips rule R1 match from-zone Zone-1 to-zone
   Zone-2 source-address any destination-address any application default
set security idp idp-policy base-policy rulebase-ips rule R1 match attacks
   predefined-attack-groups "HTTP - Critical"
set security idp idp-policy base-policy rulebase-ips rule R1 then action mark-diffserv 50
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure DSCP values in an IDP policy:

1. Create a policy by assigning a meaningful name to it.

```
[edit]
user@host# edit security idp idp-policy base-policy
```

2. Associate a rulebase with the policy.

```
[edit security idp idp-policy base-policy]
user@host# edit rulebase-ips
```

3. Add rules to the rulebase.

```
[edit security idp idp-policy base-policy rulebase-ips]
user@host# edit rule R1
```

4. Define the match criteria for the rule.

   ```
   [edit security idp idp-policy base-policy rulebase-ips R1]
   user@host# set match from-zone trust to-zone untrust source-address any
     destination-address any application default
   
   user@host# set match attacks predefined-attack-group “HTTP - Critical”
   ```

5. Specify an action for the rule.

   ```
   [edit security idp idp-policy base-policy rulebase-ips R1]
   user@host# set then action mark-diffserv 50
   ```

6. Continue to specify any notification or logging options for the rule, if required.

7. Activate the policy.

   ```
   [edit]
   user@host# set security idp active-policy base-policy
   ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy base-policy{
  rulebase-ips {
    rule R1 {
      match {
        from-zone trust;
        source-address any;
        to-zone untrust;
        destination-address any;
        application default;
        attacks {
          predefined-attack-groups HTTP-Critical;
        }
      }
      then {
        action {
          mark-diffserv {
            50;
          }
        }
      }
    }
  }
active-policy base-policy;
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 78

Verifying the Configuration

Purpose Verify that the DSCP values were configured in an IDP policy.

Action From operational mode, enter the **show security idp status** command.

Related Documentation

- IDP Policies Overview on page 43
- Intrusion Detection and Prevention Overview on page 27

Attack Objects and Object Groups for IDP Policies

Supported Platforms SRX Series, vSRX

Attack objects, application signatures objects, and service objects are used in defining IDP policy rules. As a response to new vulnerabilities, Juniper Networks periodically provides a file containing attack database updates on the Juniper website. You can download this file to protect your network from new threats. These attack objects and groups are designed to detect known attack patterns and protocol anomalies within the network traffic. You can configure attack objects and groups as match conditions in IDP policy rules.

For more information, see the following topics:

- Understanding Our Approach to Addressing Known and Unknown Vulnerabilities on page 79
- Testing a Custom Attack Object on page 80
- Creating a Signature Attack Object on page 81
- Understanding Predefined IDP Attack Objects and Object Groups on page 92
- Understanding Custom Attack Objects on page 93
- IDP Custom Attack Objects Service Contexts on page 106
- Creating a Compound Attack Object on page 161
- Modifying Custom Attack Objects Due to Changes Introduced in Signature Update on page 162
- Example: Configuring Compound or Chain Attacks on page 165
- Example: Configuring Attack Groups with Dynamic Attack Groups and Custom Attack Groups on page 171
- Custom Attack Object DFA Expressions on page 177
- Example: Using Pattern Negation on page 179

Understanding Our Approach to Addressing Known and Unknown Vulnerabilities

This topic includes the following sections:

Known Vulnerabilities

Known vulnerabilities are those documented within the Internet security community. The Internet security community comprises several security organizations, security analysts, and security forums. The security community continually discovers and analyzes new attacks and exchanges this information over the Internet. In this way, they can quickly locate, identify, and truly understand an attack.

Some security advisories include the actual attack code. You can use the attack information and the attack code to capture packet information and service contexts. You can use this information to create a custom signature attack object.

Unfortunately, most advisories do not post the attack code with the attack description. If you cannot obtain the attack code, read the advisory carefully and try to reconstruct the basics of the attack packet.

CAUTION: Remember to isolate code acquired from unknown sources.

The following organizations are active in the security community and are a good resource for locating attack information:

- NVD—National Vulnerability Database (http://nvd.nist.gov). The U.S. government repository of vulnerability management data represented using the Security Content Automation Protocol (SCAP).
- SANS—SysAdmin, Audit, Network, Security Institute (www.sans.org). An information security research, certification, and education organization that provides security alerts. Also hosts the Internet Storm Center (ISC) at http://www.incidents.org.
- CVE—Common Vulnerabilities and Exposures (http://cve.mitre.org). A standardized list of vulnerabilities and other information security exposures.
- BugTraq (http://securityfocus.com/archive/1). A moderated mailing list hosted by Security Focus that discusses and announces computer security vulnerabilities.
- CERT coordination center (http://www.cert.org). A federally funded security alert organization that provides security advisories.
- Packet Storm Security (http://packetstormsecurity.nl). A nonprofit organization of security professionals that provides security information by way of security news, advisories, forums, and attack code.
- Metasploit (http://www.metasploit.com). Metasploit provides useful information for performing penetration testing, IDS signature development, and exploit research.
- FrSIRT—French Security Incident Response Team (http://www.frsirt.com). FrSIRT is an independent security research organization providing security advisories and real-time vulnerability alerting and notification services.
- ISS—Internet Security Systems (http://www.iss.net). An Internet security company that provides alerts and Internet threat levels.

Unknown Vulnerabilities

Unknown vulnerabilities are those that have not been documented in Internet security community advisories. In these cases, the IDP Series Profiler, firewall, or IDP security event logs generated in your production environment alert you to suspicious activity and abnormal traffic. In your production environment, you will use packet logging tools to capture packets and service context information that you can later analyze and experiment with in your lab.

Testing a Custom Attack Object

We recommend the following workflow to test a custom attack object. Note that the following procedure consists of general steps and is intended for expert users who are familiar with these tasks.

To test a custom attack object:

1. Create a new security policy and new IDP rulebase rule that includes only the custom attack object to be tested. Enable logging and packet logging.

2. Push the policy to the IDP Series lab device.

3. From the attacker computer, reproduce the attack that targets the victim computer.

4. Use the Security Director Log Viewer to see whether the traffic generated logs as expected.

If your test fails, review the attack advisory, the protocol RFC, and the attack code or packet captures to identify additional information that can help you fine-tune your settings. The most frequent issue that requires tuning is the syntax of the DFA expression.

Creating a Signature Attack Object

A signature attack object is a pattern you want the system to detect. You use a DFA expression to represent the pattern. All of the other signature properties you can set (such as service or protocol context, direction, and other constraints) are provided so you can optimize performance of the system in detecting the pattern and eliminate false positives. In general, you want to tune settings of a signature attack object so that the system looks for it in every context where it might occur and in no other context.

To configure a signature attack object:

1. In the Object Manager, select **Attack Objects > IDP Objects**.

2. Click the **Custom Attacks** tab.

3. Click the + icon to display the Custom Attack dialog box.

4. Configure attack object settings. Table 10 on page 81 provides guidelines for completing the settings.

Table 10: Custom Attack Dialog Box: General Tab Settings

Setting	Description
Name	The name displayed in the UI. TIP: Include the protocol the attack uses as part of the attack name.
Description	(Optional) Information about the attack. Although a description is optional when you create a new attack object, it can help you remember important information about the attack. For examples, view the attack descriptions for predefined attacks.

Table 10: Custom Attack Dialog Box: General Tab Settings (continued)

Setting	Description
Severity	Info, Warning, Minor, Major, or Critical. Critical attacks are attempts to crash your server or gain control of your network. • Critical—Contains attack objects matching exploits that attempt to evade detection, cause a network device to crash, or gain system-level privileges. • Info—Contains attack objects matching normal, harmless traffic containing URLs, DNS lookup failures, SNMP public community strings, and peer-to-peer (P2P) parameters. You can use informational attack objects to obtain information about your network. • Major—Contains attack objects matching exploits that attempt to disrupt a service, gain user-level access to a network device, or activate a Trojan horse previously loaded on a device. • Minor—Contains attack objects matching exploits that detect reconnaissance efforts attempting to access vital information through directory traversal or information leaks. • Warning—Contains attack objects matching exploits that attempt to obtain noncritical information or scan a network with a scanning tool. Informational attacks are the least dangerous and typically are used by network administrators to discover holes in their own security system.
Category	A predefined or new category. Use this category to group the attack objects. Within each category, attack objects are grouped by severity. For example: FTP, TROJAN, SNMP.
Keywords	Unique identifiers that can be used to search and sort log records. Keywords should related to the attack and the attack object.
Recommended	Indicates that this attack object is among your highest-risk set of attack objects. Later, when you add this attack object to dynamic groups, you can specify whether to include only recommended attack objects. • Yes—Adds predefined attacks recommended by Juniper Networks to the dynamic group. • No—Specifies non-recommended attack objects in the dynamic attack group.
Detection Performance	Specify this filter to filter out slow-performing attack objects. You can use this filter to only select the appropriate attacks based on performance impacts. Select an option: • High—Add a high performance impact attack object that is vulnerable to an attack. The performance impact of signatures is high7 to high9, where the application identification is slow. • Medium—Add a medium performance impact attack object that is vulnerable to an attack. The performance impact of signatures is medium4 to medium6, where the application identification is normal. • Low—Add a low performance impact attack object that is vulnerable to an attack. The performance impact of signatures is low1 to low3, where the application identification is faster. • Unknown—Set all attack objects to unknown by default. As you fine-tune IPS to your network traffic, you can change this setting to help you track performance impact. The performance impact of signatures is 0 = unknown, where the application identification is also unknown.

5. Click the **General** tab.

6. Under Attack Versions, click the + icon to display the New Attack wizard.

7. On the Target Platform and Type page, select a device platform and attack type. Table 11 on page 83 describes the attack types.

Table 11: Attack Object Types

Type	Description
Signature	Uses a stateful attack signature (a pattern that always exists within a specific section of the attack) to detect known attacks. Stateful signature attack objects also include the protocol or service used to perpetrate the attack and the context in which the attack occurs. If you know the exact attack signature, the protocol, and the attack context used for a known attack, select this option.
Compound Attack	Detects attacks that use multiple methods to exploit a vulnerability. This object combines multiple signatures or protocol anomalies into a single attack object, forcing traffic to match all combined signatures or anomalies within the compound attack object before traffic is identified as an attack. By combining and even specifying the order in which signatures or anomalies must match, you can be very specific about the events that must place before the IDP engine identifies traffic as an attack. If you need to detect an attack that uses several benign activities to attack your network, or if you want to enforce a specific sequence of events to occur before the attack is considered malicious, select this option.

8. Select **Signature** and click **Next**.

9. On the Custom Attack – General Properties page, configure other settings. Table 12 on page 83 provides guidelines for completing the settings.

Table 12: Custom Attack – General Properties

Property	Description
Signature Details	
Binding	**Service**–If you were able to determine the service through your research, select **Service**. Later in the wizard, you can specify a service context.
	IP–If you are not sure of the service but you know IP details, select **IP** and specify a protocol type number.
	TCP, **UDP**, or **ICMP**–If you do not know the service context but you know protocol details, select the protocol. For TCP and UDP protocol types, specify the port ranges.
	RPC–If you are detecting threats over remote procedure call (RPC) protocol, select this option and specify the program ID. RPC is used by distributed processing applications to handle interaction between processes remotely. When a client makes a remote procedure call to an RPC server, the server replies with a remote program. Each remote program uses a different program number.

Table 12: Custom Attack – General Properties (continued)

Enable	Time binding attributes track how many times a signature is repeated. By configuring the scope and count of an attack, you can detect a sequence of the same attacks over a period of time (one minute) across sessions. This method is useful for detecting brute force attacks that attempt to guess authentication credentials or overwhelm system capacity to handle data.
Service	Specify the service that the attack uses to enter your network. You can select the specific service used to perpetrate the attack as the service binding. For example, suppose you select the DISCARD service. Discard protocol is an Application Layer protocol where TCP/9, UDP/9 describes the process for discarding TCP or UDP data sent to port 9.
Time Scope	Select the scope within which the count occurs: • **Source IP**–Detects the signature in traffic from the source IP address for the specified number of times, regardless of the destination IP address. • **Destination IP**–Detects the signature in traffic from the destination IP address for the specified number of times, regardless of the source IP address. • **Peer**–Detects the signature in traffic between source and destination IP addresses of the sessions for the specified number of times.
Time Count	Specify the number of times that the attack object must detect an attack within the specified scope before the device considers the attack object to match the attack. The range is from 0 through 4,294,967,295.
Match Assurance	Specify this filter to track attack objects based on the frequency that the attack produces a false positive on your network. Select an option: • **High**—Provides information on the frequently tracked false positive occurrences. • **Medium**—Provides information on the occasionally tracked false positive occurrences. • **Low**—Provides information on the rarely tracked false positive occurrences.
Performance Impact	Specify this filter to filter out slow-performing attack objects. You can use this filter to only select the appropriate attacks based on performance impacts. Select an option: • **High**—Add a high performance impact attack object that is vulnerable to an attack. The performance impact of signatures is high7 to high9, where the application identification is slow. • **Medium**—Add a medium performance impact attack object that is vulnerable to an attack. The performance impact of signatures is medium4 to medium6, where the application identification is normal. • **Low**—Add a low performance impact attack object that is vulnerable to an attack. The performance impact of signatures is low1 to low3, where the application identification is faster. • **Unknown**—Set all attack objects to unknown by default. As you fine-tune IPS to your network traffic, you can change this setting to help you track performance impact. The performance impact of signatures is 0 = unknown, where the application identification is also unknown.
Scope	Specify if the attack is matched within a session or across transactions in a session: • **session**—Allows multiple matches for the object within the same session. • **transaction**—Matches the object across multiple transactions that occur within the same session.

Click **Next**.

10. On the Custom Attack – Attack Pattern page, configure pattern settings. Table 13 on page 85 provides guidelines for completing the settings.

Table 13: Custom Attack – Attack Pattern

Setting	Description
Pattern	A DFA expression. The following rows summarize DFA syntax conventions. For detailed information, consult a standard source on programming with regular expressions.

Table 13: Custom Attack – Attack Pattern (continued)

Setting	Description	
	\B.0.1..00\B	Bit-level matching for binary protocols. The length of the bitmask must be in multiples of 8. The first \B denotes the start of the bitmask. The last \B denotes the end of the bitmask. The decimal (.) indicates the bit can be either 0 or 1. A 0 or 1 indicates the bit at that position must be 0, or must be 1.
	\0 <octal_number>	For a direct binary match.
	\X<hexadecimal-number>\X	For a direct binary match.
	\[<character-set>\]	For case-insensitive matches.
	.	To match any symbol.
	*	To match 0 or more symbols.
	+	To match 1 or more symbols.
	?	To match 0 or 1 symbol.
	()	Grouping of expressions.
	|	Alternation. Typically used with (). Example: The following expression matches dog or cat: (dog | cat).
	[]	Character class. Any explicit value within the bracket at the position matches. Example: [Dd]ay matches Day and day.
	[<start>-<end>]	Character range. Any value within the range (denoted with a hyphen). You can mix character class and a hexadecimal range. Example: [AaBbCcDdEeFf0-9].
	[^<start>-<end>]	Negation of character range. Example: [^Dd]ay matches Hay and ray, but not Day or day. NOTE: To negate an entire signature pattern, select the Negate option under the pattern text box.
	\u<string>\u	Unicode insensitive matches.
	\s	Whitespace.

Table 13: Custom Attack – Attack Pattern (continued)

<table>
<tr><th>Setting</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>\</td><td>Use a backslash to escape special characters so that they are matched and not processed as regular expression operators.
<table>
<tr><th>Character</th><th>Escaped</th></tr>
<tr><td>*</td><td>*</td></tr>
<tr><td>(</td><td>\(</td></tr>
<tr><td>)</td><td>\)</td></tr>
<tr><td>.</td><td>\.</td></tr>
<tr><td>+</td><td>\+</td></tr>
<tr><td>\</td><td>\\</td></tr>
<tr><td>[</td><td>\0133</td></tr>
<tr><td>]</td><td>\0135</td></tr>
</table>
NOTE: Because the combination of the backslash and the open and close square brackets are used in the case-insensitive expression, you must use the backslash with the octal code for the bracket characters.</td></tr>
</table>
</td></tr>
<tr><td>Negate</td><td>Negates the attack pattern.</td></tr>
<tr><td>Regex</td><td>Enter a regular expression to define rules to match malicious or unwanted behavior over the network.
For example: For the syntax \[hello\], the expected pattern is hello, which is case sensitive.
The example matches can be: hElLo, HEllO, and heLLO.</td></tr>
</table>

Table 13: Custom Attack – Attack Pattern (continued)

Setting	Description
Context	Binds pattern matching to a context. For known services, such as HTTP, select the service in the first box, and select the HTTP context you discovered with **scio ccap**, such as HTTP POST Parsed Param, in the second box. If you were unable to discover the context, select **Other** in the first box, and select one of the following contexts in the second box: • **Packet**–Detects the pattern in any packet. • **First Packet**–Inspects only the first packet of a stream. When the flow direction is set to any, the detector engine checks the first packet of both the server-to-client (STC) and client-to-server (CTS) flows. Less processing means greater performance. If you know that the pattern appears in the first packet of a session, select **First Packet**. • **First Data Packet**–Inspection ends after the first packet of a stream. Select this option to detect the attack in only the first data packet of a stream. If you know that the pattern appears in the first data packet of a stream, select **First Data Packet**. • **Stream 256**–Reassembles packets and searches for a pattern match within the first 256 bytes of a traffic stream. Stream 256 is often the best choice for non-UDP attacks. When the flow direction is set to **any**, the detector engine checks the first 256 bytes of both the STC and CTS flows. If you know that the pattern will appear in the first 256 bytes of a session, select **Stream 256**. • **Stream 8K**–Like Stream 256 except reassembles packets and searches for a pattern match within the first 8192 bytes of a traffic stream. • **Stream 1K**–Like Stream 256 except reassembles packets and searches for a pattern match within the first 1024 bytes of a traffic stream. • **Line**–Detects a pattern within a specific line. Use this context for line-oriented applications or protocols (such as FTP). • **Stream**–Reassembles packets and extracts the data to search for a pattern match. However, the IDP engine does not recognize packet boundaries for stream contexts, so data for multiple packets is combined. Select this option only when no other context option contains the attack. NOTE: If you select a line, stream, or service context, you do not configure match criteria for IP settings and protocol header fields.
Direction	Select the direction in which to detect the pattern: • **Client to Server**–Detects the pattern only in client-to-server traffic. • **Server to Client**–Detects the pattern only in server-to-client traffic. • **Any**–Detects the pattern in either direction. The session initiator is considered the client, even if that source IP is a server.
Add Anomaly	
Anomaly	Select an option to detect abnormal or ambiguous messages within a connection according to the set of rules for the particular protocol being used. Protocol anomaly detection works by finding deviations from protocol standards, most often defined by RFCs and common RFC extensions.

Table 13: Custom Attack – Attack Pattern (continued)

Setting	Description
Direction	Specify the connection direction of the attack: • **Client to Server**—Detects the attack only in client-to-server traffic. • **Server to Client**—Detects the attack only in server-to-client traffic. • **Any**—Detects the attack in either direction. Using a single direction (instead of Any) improves performance, reduces false positives, and increases detection accuracy.

Click **Next**.

11. If you have selected a line, stream, stream 256, or service context, do not configure match criteria for IP settings and protocol header fields. Click **Finish**.

    If you are using a packet context, you can refine matching by adding criteria for IP flags and packet headers, as described in the following tables.

    TIP: If you are unsure of the IP flags and IP fields you want to match, leave all fields blank. If no values are set, the IDP engine attempts to match the signature for all header contents.

    On the Custom Attack – IPv4 settings and header matches page, configure pattern settings. Table 14 on page 89 provides guidelines for completing the settings.

Table 14: Custom Attack – IPv4 Settings and Header Matches Page

Setting	Description
Checksum Validate	Validate checksum field against calculated checksum.
Type of Service	Service type. Common service types are: • 0000 Default • 0001 Minimize Cost • 0002 Maximize Reliability • 0003 Maximize Throughput • 0004 Minimize Delay • 0005 Maximize Security
IP Flags	IP Flag bits.
IHL	Internet header length in words.
Total Length	Total Length of IP datagram.
ID	Unique value used by the destination system to reassemble a fragmented packet.

Table 14: Custom Attack – IPv4 Settings and Header Matches Page (continued)

Setting	Description
Time-to-live	Time-to-live (TTL) value of the packet. This value represents the number of routers the packet can pass through. Each router that processes the packet decrements the TTL by 1; when the TTL reaches 0, the packet is discarded.
Protocol	Protocol used in the attack.
Source	IP address of the attacking device.
Destination	IP address of the attack target.

On the Custom Attack – IPv6 settings and header matches page, configure pattern settings. Table 15 on page 90 provides guidelines for completing the settings.

Table 15: Custom Attack – IPv6 Settings and Header Matches Page

Setting	Description
Destination	IP address of the attack target.
Extension Header	Define the IPv6 extension header for the intrusion detection service (IDS).
Flow Label	Enable IPv6 packet flow labels.
Hop Limit	Specifies the maximum number of hops that the router can use in router advertisements and all IPv6 packets.
Next Header	Identifies the type of Internet Protocol for the header that immediately follows the IPv6 header.
Payload Length	Specifies the length of the IPv6 packet payload, or contents, expressed in octets.
Source	Identifies the host device, or interface on a node, that generated the IPv6 packet.
Traffic Class	Allows source nodes or routers to identify different classes (or priorities for quality of service) for IPv6 packets. (This field replaces the IPv4 Type of Service field.)

On the Custom Attack – TCP packet header page, configure pattern settings. Table 16 on page 90 provides guidelines for completing the settings.

Table 16: Custom Attack Object: TCP Packet Header Fields

Setting	Description
Source Port	Port number on the attacking device.
Destination Port	Port number of the attack target.
Sequence Number	Sequence number of the packet. This number identifies the location of the data in relation to the entire data sequence.

Table 16: Custom Attack Object: TCP Packet Header Fields (continued)

Setting	Description
ACK Number	ACK number of the packet. This number identifies the next sequence number; the ACK flag must be set to activate this field.
Header Length	Number of bytes in the TCP header.
Window Size	Number of bytes in the TCP window size.
Data Length	Number of bytes in the data payload. For SYN, ACK, and FIN packets, this field should be empty.
Urgent Pointer	Data in the packet is urgent; the URG flag must be set to activate this field.
MSS	Enable and specify the TCP maximum segment size.
Reserved	Specify the three reserved bits in the TCP header field.
TCP Flags	TCP header flags. Specify that IDP looks for a pattern match whether or not the TCP flag is set.
Window Scale	Specify the scale factor that the session of the attack will use.

On the Custom Attack – UDP header page, configure pattern settings. Table 17 on page 91 provides guidelines for completing the settings.

Table 17: Custom Attack Object: UDP Header Fields

Setting	Description
Checksum Validate	Validate checksum field against calculated checksum.
Source Port	Port number on the attacking device.
Destination Port	Port number of the attack target.
Data Length	Number of bytes in the data payload.

On the Custom Attack – ICMP packet header page, configure pattern settings. Table 18 on page 91 provides guidelines for completing the settings.

Table 18: Custom Attack Object: ICMP Packet Header Fields

Setting	Description
Checksum Validate	Validate checksum field against calculated checksum.
ICMP Type	Primary code that identifies the function of the request or reply.
ICMP Code	Secondary code that identifies the function of the request or reply within a given type.

Table 18: Custom Attack Object: ICMP Packet Header Fields (continued)

Setting	Description
Sequence Number	Sequence number of the packet. This number identifies the location of the request/reply in relation to the entire sequence.
ICMP ID	Identification number, which is a unique value used by the destination system to associate requests and replies.
Data length	Number of bytes in the data payload.

12. Click **Finish**.

Understanding Predefined IDP Attack Objects and Object Groups

The security package for Intrusion Detection and Prevention (IDP) contains a database of predefined IDP attack objects and IDP attack object groups that you can use in IDP policies to match traffic against known and unknown attacks. Juniper Networks updates the predefined attack objects and groups on a regular basis with newly discovered attack patterns.

Updates to the attack object database can include:

- New descriptions or severities for existing attack objects
- New attack objects
- Deletion of obsolete attack objects

This topic includes the following sections:

- Predefined Attack Objects on page 92
- Predefined Attack Object Groups on page 92

Predefined Attack Objects

Predefined attack objects are listed in an alphabetical order. These attack objects have unique names that help you identify the attack. The first part of the name indicates the group to which the attack object belongs. For example:

- **FTP:USER:ROOT**—Belongs to the **FTP:USER** group. It detects attempts to log in to an FTP server using the **root** account.
- **HTTP:HOTMAIL:FILE-UPLOAD**—Belongs to the **HTTP:HOTMAIL** group. It detects files attached to e-mails sent via the Web-based e-mail service **Hotmail**.

Predefined Attack Object Groups

The predefined attack groups list displays the attack objects in the categories described below. A set of recommended attack objects that Juniper Networks considers to be serious threats are also available in this list. The recommended attack objects are organized into the following categories:

Table 19: Predefined Attack Object Groups

Attack Object Group	Description
Attack Type	Groups attack objects by type (anomaly or signature). Within each type, attack objects are grouped by severity.
Category	Groups attack objects by predefined categories. Within each category, attack objects are grouped by severity.
Operating System	Groups attack objects by the operating system to which they apply: BSD, Linux, Solaris, or Windows. Within each operating system, attack objects are grouped by services and severity.
Severity	Groups attack objects by the severity assigned to the attack. IDP has five severity levels: Critical, Major, Minor, Warning, Info. Within each severity, attack objects are grouped by category.
Web Services	Groups attack objects by common Web services. These services are grouped by severity levels—Warning, Critical, Major, Minor, Info.
Miscellaneous	Groups attack objects by performance level. Attack objects affecting IDP performance over a certain level are grouped under this category.
Response	Groups attack objects in traffic flowing in the server to client direction.

Understanding Custom Attack Objects

You can create custom attack objects to detect new attacks or customize predefined attack objects to meet the unique needs of your network.

To configure a custom attack object, you specify a unique name for it and then specify additional information, such as a general description and keywords, which can make it easier for you to locate and maintain the attack object.

Certain properties in the attack object definitions are common to all types of attacks, such as attack name, description, severity level, service or application binding, time binding, recommended action, and protocol or port binding. Some fields are specific to an attack type and are available only for that specific attack definition.

NOTE: IDP feature is enabled by default, no license is required. Custom attacks and custom attack groups in IDP policies can also be configured and installed even when a valid license and signature database are not installed on the device.

This topic includes the following sections:

Attack Name

Specify an alphanumeric name for the object. You might want to include the protocol the attack uses in the attack name.

Starting with Junos OS Release 15.1X49-D140, the maximum number of characters allowed for a custom attack object name is 60. You can validate the statement using the **set security idp custom-attack** command.

Severity

Specifies the brutality of the attack on your network. Severity categories, in order of increasing brutality, are info, warning, minor, major, critical. Critical attacks are the most dangerous—typically these attacks attempt to crash your server or gain control of your network. Informational attacks are the least dangerous, and typically are used by network administrators to discover holes in their own security systems.

Service and Application Bindings

The service or application binding field specifies the service that the attack uses to enter your network.

NOTE: **Specify either the service or the protocol binding in a custom attack. In case you specify both, the service binding takes precedence.**

- **any**—Specify **any** if you are unsure of the correct service and want to match the signature in all services. Because some attacks use multiple services to attack your network, you might want to select the **Any** service binding to detect the attack regardless of which service the attack chooses for a connection.
- **service**—Most attacks use a specific service to attack your network. You can select the specific service used to perpetrate the attack as the service binding.

  For list of services, service bindings ,and contexts see “IDP Custom Attack Objects Service Contexts” on page 106

Protocol and Port Bindings

Protocol or port bindings allow you to specify the protocol that an attack uses to enter your network. You can specify the name of the network protocol or the protocol number.

NOTE: Specify either the service or the protocol binding in a custom attack. In case you specify both, the service binding takes precedence.

- IP—You can specify any of the supported network layer protocols using protocol numbers. Table 20 on page 95 lists protocol numbers for different protocols.

Table 20: Supported Protocols and Protocol Numbers

Protocol Name	Protocol Number
IGMP	2
IP-IP	4
EGP	8
PUP	12
TP	29
IPV6	41
ROUTING	43
FRAGMENT	44
RSVP	46
GRE	47
ESP	50
AH	51
ICMPV6	58
NONE	59
DSTOPTS	60
MTP	92
ENCAP	98

Table 20: Supported Protocols and Protocol Numbers (continued)

Protocol Name	Protocol Number
PIM	103
COMP	108
RAW	255

- ICMP, TCP, and UDP—Attacks that do not use a specific service might use specific ports to attack your network. Some TCP and UDP attacks use standard ports to enter your network and establish a connection.
- RPC—The remote procedure call (RPC) protocol is used by distributed processing applications to handle interaction between processes remotely. When a client makes a remote procedure call to an RPC server, the server replies with a remote program; each remote program uses a different program number. To detect attacks that use RPC, configure the service binding as RPC and specify the RPC program ID.

Table 21 on page 96 displays sample formats for key protocols.

Table 21: Sample Formats for Protocols

Protocol Name	Protocol Number	Description
ICMP	<Port>ICMP</Port>	Specify the protocol name.
IP	<Port>IP/*protocol-number*</Port>	Specify the Network Layer protocol number.
RPC	<Port>RPC/*program-number*</Port>	Specify the RPC program number.
TCP or UDP	• <Port>TCP </Port> • <Port>TCP/*port* </Port> • <Port>TCP/*minport-maxport* </Port>	Specifying the port is optional for TCP and UDP protocols. For example, you can specify any of the following: • <Port>UDP</Port> • <Port>UDP/10</Port> • <Port>UDP/10-100</Port>

Time Bindings

Use time bindings to configure the time attributes for the custom attack object. Time attributes control how the attack object identifies attacks that repeat for a certain number of times. By configuring the scope and count of an attack, you can detect a sequence of the same attacks over a period of time (one minute) across sessions.

Scope

Specify the scope within which the count of an attack occurs:

- Source—Specify this option to detect attacks from the source address for the specified number of times, regardless of the destination address. This means that for a given attack, a threshold value is maintained for each attack from the source address. The destination address is ignored. For example, anomalies are detected from two different pairs (**ip-a**, **ip-b**) and (**ip-a**, **ip-c**) that have the same source address **ip-a** but different destination addresses **ip-b** and **ip-c**. Then the number of matches for **ip-a** increments to **2**. Suppose the threshold value or *count* is also set to 2, then the signature triggers the attack event.
- Destination—Specify this option to detect attacks sent to the destination address for the specified number of times, regardless of the source address. This means that for a given attack, a threshold value is maintained for each attack from the destination address. The source address is ignored. For example, if anomalies are detected from two different pairs (**ip-a**, **ip-b**) and (**ip-c**, **ip-b**) that have the same destination address **ip-b** but different source addresses **ip-a** and **ip-c**. Then the number of matches for **ip-b** increments to **2**. Suppose the threshold value or *count* is also set to **2**, then the signature triggers the attack event.
- Peer—Specify this option to detect attacks between source and destination IP addresses of the sessions for the specified number of times. This means that the threshold value is applicable for a pair of source and destination addresses. Suppose anomalies are detected from two different source and destination pairs (**ip-a**, **ip-b**) and (**ip-a**, **ip-c**). Then the number of matches for each pair is set to **1**, even though both pairs have a common source address.

Count

Count or threshold value specifies the number of times that the attack object must detect an attack within the specified scope before the device considers the attack object to match the attack. If you bind the attack object to multiple ports and the attack object detects that attack on different ports, each attack on each port is counted as a separate occurrence. For example, when the attack object detects an attack on **TCP/80** and then on **TCP/8080**, the count is two.

Once the **count** match is reached, each attack that matches the criteria causes the attack count to increase by one. This count cycle lasts for a duration of 60 seconds, after which the cycle repeats.

Attack Properties (Signature Attacks)

Signature attack objects use a stateful attack signature (a pattern that always exists within a specific section of the attack) to detect known attacks. They also include the protocol or service used to perpetrate the attack and the context in which the attack occurs. The following properties are specific to signature attacks, and you can configure them when configuring signature attack:

NOTE: Attack context, flow type, and direction are mandatory fields for the signature attack definition.

Attack Context

An attack context defines the location of the signature. If you know the service and the specific service context, specify that service and then specify the appropriate service contexts. If you know the service, but are unsure of the specific service context, specify one of the following general contexts:

- **first-data-packet**—Specify this context to detect the attack in only the first data packet.
- **first-packet**—Specify this context to detect the attack in only the first packet of a stream. When the flow direction for the attack object is set to **any**, the device checks the first packet of both the server-to-client and the client-to-server flows. If you know that the attack signature appears in the first packet of a session, choosing **first packet** instead of **packet** reduces the amount of traffic the device needs to monitor, which improves performance.
- **packet**—Specify this context to match the attack pattern within a packet. When you select this option, you must also specify the service binding to define the service header options . Although not required, specifying these additional parameters improves the accuracy of the attack object and thereby improves performance.
- **line**—Specify this context to detect a pattern match within a specific line within your network traffic.
- **normalized-stream**—Specify this context to detect the attack in an entire normalized stream. The normalized stream is one of the multiple ways of sending information. In this stream the information in the packet is normalized before a match is performed. Suppose **www.yahoo.com/sports** is the same as **www.yahoo.com/s%70orts**. The normalized form to represent both of these URLs might be **www.yahoo.com/sports**. Choose **normalized stream** instead of **stream**, unless you want to detect some pattern in its exact form. For example, if you want to detect the exact pattern **www.yahoo.com/s%70orts**, then select **stream**.
- **normalized-stream256**—Specify this context to detect the attack in only the first 256 bytes of a normalized stream.
- **normalized-stream1k**—Specify this context to detect the attack in only the first 1024 bytes of a normalized stream.
- **normalized-stream-8k**—Specify this context to detect the attack in only the first 8192 bytes of a normalized stream.
- **stream**—Specify this context to reassemble packets and extract the data to search for a pattern match. However, the device cannot recognize packet boundaries for stream contexts, so data for multiple packets is combined. Specify this option only when no other context option contains the attack.
- **stream256**—Specify this context to reassemble packets and search for a pattern match within the first 256 bytes of a traffic stream. When the flow direction is set to **any**, the device checks the first 256 bytes of both the server-to-client and client-to-server flows.

If you know that the attack signature will appear in the first 256 bytes of a session, choosing **stream256** instead of **stream** reduces the amount of traffic that the device must monitor and cache, thereby improving performance.

- **stream1k**—Specify this context to reassemble packets and search for a pattern match within the first 1024 bytes of a traffic stream. When the flow direction is set to **any**, the device checks the first 1024 bytes of both the server-to-client and client-to-server flows. If you know that the attack signature will appear in the first 1024 bytes of a session, choosing **stream1024** instead of **stream** reduces the amount of traffic that the device must monitor and cache, thereby improving performance.
- **stream8k**—Specify this context to reassemble packets and search for a pattern match within the first 8192 bytes of a traffic stream. When the flow direction is set to **any**, the device checks the first 8192 bytes of both the server-to-client and client-to-server flows. If you know that the attack signature will appear in the first 8192 bytes of a session, choosing **stream8192** instead of **stream** reduces the amount of traffic that the device must monitor and cache, thereby improving performance.

Attack Direction

You can specify the connection direction of the attack. Using a single direction (instead of **Any**) improves performance, reduces false positives, and increases detection accuracy.

- Client to server (detects the attack only in client-to-server traffic)
- Server to client (detects the attack only in server-to-client traffic)
- Any (detects the attack in either direction)

Attack Pattern

Attack patterns are signatures of the attacks you want to detect. A signature is a pattern that always exists within an attack; if the attack is present, so is the signature. To create the attack pattern, you must first analyze the attack to detect a pattern (such as a segment of code, a URL, or a value in a packet header), then create a syntactical expression that represents that pattern. You can also negate a pattern. Negating a pattern means that the attack is considered matched if the pattern defined in the attack does *not* match the specified pattern.

NOTE: **Pattern negation is supported for packet, line, and application based contexts only and not for stream and normalized stream contexts.**

Protocol-Specific Parameters

Specifies certain values and options existing within packet headers. These parameters are different for different protocols. In a custom attack definition, you can specify fields for only one of the following protocols—TCP, UDP, or ICMP. Although, you can define IP protocol fields with TCP or UDP in a custom attack definition.

NOTE: Header parameters can be defined only for attack objects that use a packet or first packet context. If you specified a line, stream, stream 256, or a service context, you cannot specify header parameters.

If you are unsure of the options or flag settings for the malicious packet, leave all fields blank and Intrusion Detection and Prevention (IDP) attempts to match the signature for all header contents.

Table 22 on page 100 displays fields and flags that you can set for attacks that use the IP protocol.

Table 22: IP Protocol Fields and Flags

Field	Description
Type of Service	Specify a value for the service type. Common service types are: • 0000 Default • 0001 Minimize Cost • 0002 Maximize Reliability • 0003 Maximize Throughput • 0004 Minimize Delay • 0005 Maximize Security
Total Length	Specify a value for the number of bytes in the packet, including all header fields and the data payload.
ID	Specify a value for the unique value used by the destination system to reassemble a fragmented packet.
Time to Live	Specify an integer value in the range of 0–255 for the time-to-live (TTL) value of the packet. This value represents the number of devices the packet can traverse. Each router that processes the packet decrements the TTL by 1; when the TTL reaches 0, the packet is discarded.
Protocol	Specify a value for the protocol used.
Source	Enter the source address of the attacking device.
Destination	Enter the destination address of the attack target.
Reserved Bit	This bit is not used.
More Fragments	When set (1), this option indicates that the packet contains more fragments. When unset (0), it indicates that no more fragments remain.
Don’t Fragment	When set (1), this option indicates that the packet cannot be fragmented for transmission.

Table 23 on page 101 displays packet header fields and flags that you can set for attacks that use the TCP protocol.

Table 23: TCP Header Fields and Flags

Field	Description
Source Port	Specify a value for the port number on the attacking device.
Destination Port	Specify a value for the port number of the attack target.
Sequence Number	Specify a value for the sequence number of the packet. This number identifies the location of the data in relation to the entire data sequence.
ACK Number	Specify a value for the ACK number of the packet. This number identifies the next sequence number; the ACK flag must be set to activate this field.
Header Length	Specify a value for the number of bytes in the TCP header.
Data Length	Specify a value for the number of bytes in the data payload. For SYN, ACK, and FIN packets, this field should be empty.
Window Size	Specify a value for the number of bytes in the TCP window size.
Urgent Pointer	Specify a value for the urgent pointer. The value indicates that the data in the packet is urgent; the URG flag must be set to activate this field.
URG	When set, the urgent flag indicates that the packet data is urgent.
ACK	When set, the acknowledgment flag acknowledges receipt of a packet.
PSH	When set, the push flag indicates that the receiver should push all data in the current sequence to the destination application (identified by the port number) without waiting for the remaining packets in the sequence.
RST	When set, the reset flag resets the TCP connection, discarding all packets in an existing sequence.
SYN	When set, the SYN flag indicates a request for a new session.
FIN	When set, the final flag indicates that the packet transfer is complete and the connection can be closed.
R1	This reserved bit (1 of 2) is not used.
R2	This reserved bit (2 of 2) is not used.

Table 24 on page 102 displays packet header fields and flags that you can set for attacks that use the UDP protocol.

Table 24: UDP Header Fields and Flags

Field	Description
Source Port	Specify a value for the port number on the attacking device.
Destination Port	Specify a value for the port number of the attack target.
Data Length	Specify a value for the number of bytes in the data payload.

Table 25 on page 102 displays packet header fields and flags that you can set for attacks that use the ICMP protocol.

Table 25: ICMP Header Fields and Flags

Field	Description
ICMP Type	Specify a value for the primary code that identifies the function of the request or reply packet.
ICMP Code	Specify a value for the secondary code that identifies the function of the request or reply packet within a given type.
Sequence Number	Specify a value for the sequence number of the packet. This number identifies the location of the request or reply packet in relation to the entire sequence.
ICMP ID	Specify a value for the identification number. The identification number is a unique value used by the destination system to associate request and reply packets.
Data Length	Specify a value for the number of bytes in the data payload.

Sample Signature Attack Definition

The following is a sample signature attack definition:

```
<Entry>
<Name>sample-sig</Name>
<Severity>Major</Severity>
<Attacks><Attack>
<TimeBinding><Count>2</Count>
<Scope>dst</Scope></TimeBinding>
<Application>FTP</Application>
<Type>signature</Type>
<Context>packet</Context>
<Negate>true</Negate>
<Flow>Control</Flow>
<Direction>any</Direction>
<Headers><Protocol><Name>ip</Name>
<Field><Name>ttl</Name>
<Match>==</Match><Value>128</Value></Field>
</Protocol><Name>tcp</Name>
```

```
<Field><Name><Match>&lt;</Match>
<value>1500</Value>
</Field></Protocol></Headers>
</Attack></Attacks>
</Entry>
```

Attack Properties (Protocol Anomaly Attacks)

A protocol anomaly attack object detects unknown or sophisticated attacks that violate protocol specifications (RFCs and common RFC extensions). You cannot create new protocol anomalies, but you can configure a new attack object that controls how your device handles a predefined protocol anomaly when detected.

NOTE: **The service or application binding is a mandatory field for protocol anomaly attacks.**

The following properties are specific to protocol anomaly attacks. Both attack direction and test condition are mandatory fields for configuring anomaly attack definitions.

Attack Direction

Attack direction allows you to specify the connection direction of an attack. Using a single direction (instead of **Any**) improves performance, reduces false positives, and increases detection accuracy:

- Client to server (detects the attack only in client-to-server traffic)
- Server to client (detects the attack only in server-to-client traffic)
- Any (detects the attack in either direction)

Test Condition

Test condition is a condition to be matched for an anomaly attack. Juniper Networks supports certain predefined test conditions. In the following example, the condition is a message that is too long. If the size of the message is longer than the preconfigured value for this test condition, the attack is matched.

```
<Attacks>
<Attack>
<Type>anomaly</Type>
...
<Test>MESSAGE_TOO_LONG</Test>
<Value>yes</Value>
...
</Attack>
</Attacks>
```

Sample Protocol Anomaly Attack Definition

The following is a sample protocol anomaly attack definition:

```
<Entry>
<Name>sample-anomaly</Name>
<Severity>Info</Severity>
<Attacks><Attack>
```

```
<TimeBinding><Count>2</Count>
<Scope>peer</Scope></TimeBinding>
<Application>TCP</Application>
<Type>anomaly</Type>
<Test>OPTIONS_UNSUPPORTED</Test>
<Direction>any</Direction>
</Attack></Attacks>
</Entry>
```

Attack Properties (Compound or Chain Attacks)

A compound or chain attack object detects attacks that use multiple methods to exploit a vulnerability. This object combines multiple signatures and/or protocol anomalies into a single attack object, forcing traffic to match a pattern of combined signatures and anomalies within the compound attack object before traffic is identified as an attack. By combining and even specifying the order in which signatures or anomalies must match, you can be very specific about the events that need to take place before the device identifies traffic as an attack.

You must specify a minimum of 2 members (attacks) in a compound attack. You can specify up to 32 members in compound attack. Members can be either signature or anomaly attacks.

The following properties are specific to compound attacks:

Scope

Scope allows you to specify if the attack is matched within a session or across transactions in a session. If the specified service supports multiple transactions within a single session, you can also specify whether the match should occur over a single session or can be made across multiple transactions within a session:

- Specify *session* to allow multiple matches for the object within the same session.
- Specify *transaction* to match the object across multiple transactions that occur within the same session.

Order

Use ordered match to create a compound attack object that must match each member signature or protocol anomaly in the order you specify. If you do not specify an ordered match, the compound attack object still must match all members, but the attack pattern or protocol anomalies can appear in the attack in random order.

Reset

Specifies that a new log is generated each time an attack is detected within the same session. If this field is set to **no** then the attack is logged only once for a session.

Expression (Boolean expression)

Using the Boolean expression field disables the ordered match function. The Boolean expression field makes use of the member name or member index properties. The following three Boolean operators are supported along with parenthesis, which helps determine precedence:

- **or**—If either of the member name patterns match, the expression matches.
- **and**—If both of the member name patterns match, the expression matches. It does not matter which order the members appear in.
- **oand (ordered and)**—If both of the member name patterns match, and if they appear in the same order as specified in the Boolean expression, the expression matches.

Suppose you have created five signature members, labelled **s1-s5**. Suppose you know that the attack always contains the pattern **s1**, followed by either **s2** or **s3**. You also know that the attack always contains **s4** and **s5**, but their positions in the attack can vary. In this case, you might create the following Boolean expression:

((s1 oand s2) or (s1 oand s3)) and (s4 and s5)

NOTE: **You can either define an ordered match or an expression (not both) in a custom attack definition.**

Member Index

Member Index is specified in chain attacks to identify a member (attack) uniquely. In the following example, member index is used to identify the members **m01** and **m02** in the defined expression:

```
<Expression>m02 AND m01</Expression>
<Order>no</Order>
<Reset>no</Reset>
<ScopeOption/>
<Members>
<Attack>
<Member>m01</Member>
<Type>Signature</Type>
...
<Pattern><!CDATA[.*/getlatestversion]]></Pattern>
<Regex/>
</Attack>
<Attack><Member>m02</Member>
<Type>Signature</Type>
...
<Pattern><!CDATA[\[Skype\'.*]]></Pattern>
<Regex/>
</Attack>
<Attack>
```

NOTE: **When defining the expression, you must specify the member index for all members.**

Sample Compound Attack Definition

The following is a sample compound attack definition:

```
<Entry>
<Name>sample-chain</Name>
<Severity>Critical</Severity>
```

```
<Attacks><Attack>
<Application>HTTP</Application>
<Type>Chain</Type>
<Order>yes</Order>
<Reset>yes</Reset>
<Members><Attack>
<Type>Signature</Type>
<Context>packet</Context>
<Pattern><![CDATA[Unknown[]></Pattern>
<Flow>Control</Flow>
<Direction>cts</Direction>
</Attack><Attack>
<Type>anomaly</Type>
<Test>CHUNK_LENGTH_OVERFLOW</Test>
<Direction>any</Direction>
</Attack></Members>
</Attack></Attacks>
</Entry>
```

IDP Custom Attack Objects Service Contexts

The service or application binding field specifies the service that the attack uses to enter your network.

NOTE: **Specify either the service or the protocol binding in a custom attack. In case you specify both, the service binding takes precedence.**

- **any**—Specify **any** if you are unsure of the correct service and want to match the signature in all services. Because some attacks use multiple services to attack your network, you might want to select the **Any** service binding to detect the attack regardless of which service the attack chooses for a connection.
- **service**—Most attacks use a specific service to attack your network. You can select the specific service used to perpetrate the attack as the service binding.

  Table 26 on page 106 displays supported services and default ports associated with the services.

Table 26: Supported Services for Service Bindings

Service	Description	Default Port
aim	AOL Instant Messenger. America Online Internet service provider (ISP) provides Internet, chat, and instant messaging applications.	TCP/5190
bgp	Border Gateway Protocol	TCP/179
chargen	Character Generator Protocol is a UDP- or TCP-based debugging and measurement tool.	TCP/19, UDP/19
dhcp	Dynamic Host Configuration Protocol allocates network addresses and delivers configuration parameters from server to hosts.	UDP/67, UDP/68

Table 26: Supported Services for Service Bindings (continued)

Service	Description	Default Port
discard	Discard protocol is an Application Layer protocol that describes a process for discarding TCP or UDP data sent to port 9.	TCP/9, UDP/9
dns	Domain Name System translates domain names into IP addresses.	TCP/53, UDP/53
echo	Echo	TCP/7, UDP/7
finger	Finger is a UNIX program that provides information about users.	TCP/79, UDP/79
ftp	File Transfer Protocol (FTP) allows the sending and receiving of files between machines.	TCP/21, UDP/21
gGnutella	Gnutella is a public domain file sharing protocol that operates over a distributed network.	TCP/6346
gopher	Gopher organizes and displays Internet servers' contents as a hierarchically structured list of files.	TCP/70
h225ras	H.225.0/RAS (Registration, Admission, and Status)	UDP/1718, UDP/1719
http	HyperText Transfer Protocol is the underlying protocol used by the World Wide Web (WWW).	TCP/80, TCP/81, TCP/88, TCP/3128, TCP/7001 (Weblogic), TCP/8000, TCP/8001, TCP/8100 (JRun), TCP/8200 (JRun), TCP/8080, TCP/8888 (Oracle-9i), TCP/9080 (Websphere), UDP/80
icmp	Internet Control Message Protocol	
ident	Identification protocol is a TCP/IP Application Layer protocol used for TCP client authentication.	TCP/113
ike	Internet Key Exchange protocol (IKE) is a protocol to obtain authenticated keying material for use with ISAKMP.	UDP/500
imap	Internet Message Access Protocol is used for retrieving messages.	TCP/143, UDP/143
irc	Internet Relay Chat (IRC) allows people connected to the Internet to join live discussions.	TCP/6667
ldap	Lightweight Directory Access Protocol is a set of protocols used to access information directories.	TCP/389

Table 26: Supported Services for Service Bindings (continued)

Service	Description	Default Port
lpr	Line Printer Daemon protocol is a TCP-based protocol used for printing applications.	TCP/515
msn	Microsoft Network Messenger is a utility that allows you to send instant messages and talk online.	TCP/1863
msrpc	Microsoft Remote Procedure Call	TCP/135, UDP/135
mssql	Microsoft SQL is a proprietary database server tool that allows for the creation, access, modification, and protection of data.	TCP/1433, TCP/3306
mysql	MySQL is a database management system available for both Linux and Windows.	TCP/3306
nbds	NetBIOS Datagram Service application, published by IBM, provides connectionless (datagram) applications to PCs connected with a broadcast medium to locate resources, initiate sessions, and terminate sessions. It is unreliable and the packets are not sequenced.	UDP/137 (NBName), UDP/138 (NBDS)
nfs	Network File System uses UDP to allow network users to access shared files stored on computers of different types. SUN RPC is a building block of NFS.	TCP/2049, UDP/2049
nntp	Network News Transport Protocol is a protocol used to post, distribute, and retrieve USENET messages.	TCP/119
ntp	Network Time Protocol provides a way for computers to synchronize to a time reference.	UDP/123
pop3	Post Office Protocol is used for retrieving e-mail.	UDP/110, TCP/110
prtmapper	Service that runs on nodes on the Internet to map an ONC RPC program number to the network address of the server that listens for the program number.	TCP/111, UDP/111
radius	Remote Authentication Dial-In User Service application is a server program used for authentication and accounting purposes.	UDP/1812, UDP/1813
rexec	Rexec	TCP/512
rlogin	RLOGIN starts a terminal session on a remote host.	TCP/513
rsh	RSH executes a shell command on a remote host.	TCP/514
rtsp	Real-Time Streaming Protocol (RTSP) is for streaming media applications	TCP/554

Table 26: Supported Services for Service Bindings (continued)

Service	Description	Default Port
sip	Session Initiation Protocol (SIP) is an Application Layer control protocol for creating, modifying, and terminating sessions.	TCP/5060, UDP/5060
smb	Server Message Block (SMB) over IP is a protocol that allows you to read and write files to a server on a network.	TCP/139, TCP/445
smtp	Simple Mail Transfer Protocol is used to send messages between servers.	TCP/25, UDP/25
snmp	Simple Network Management Protocol is a set of protocols for managing complex networks.	TCP/161, UDP/161
snmptrap	SNMP trap	TCP/162, UDP/162
sqlmon	SQL monitor (Microsoft)	UDP/1434
ssh	SSH is a program to log into another computer over a network through strong authentication and secure communications on a channel that is not secure.	TCP/22, UDP/22
ssl	Secure Sockets Layer	TCP/443, TCP/80
syslog	Syslog is a UNIX program that sends messages to the system logger.	UDP/514
tlnet	Telnet is a UNIX program that provides a standard method of interfacing terminal routers and terminal-oriented processes to each other.	TCP/23, UDP/23
tns	Transparent Network Substrate	TCP/1521, TCP/1522, TCP/1523, TCP/1524, TCP/1525, TCP/1526, TCP/1527, TCP/1528, TCP/1529, TCP/1530, TCP/2481, TCP/1810, TCP/7778
tftp	Trivial File Transfer Protocol	UDP/69
vnc	Virtual Network Computing facilitates viewing and interacting with another computer or mobile router connected to the Internet.	TCP/5800, TCP/5900
whois	Network Directory Application Protocol is a way to look up domain names.	TCP/43
ymsg	Yahoo! Messenger is a utility that allows you to check when others are online, send instant messages, and talk online.	TCP/5050

Table 27: Service Contexts: AIM

Context and Direction	Description	Display Name
aim-auth-request-msg (ANY)	Matches the message sent from one user to another when requesting authorization to add to the buddy list.	AIM Auth Request Msg
aim-away-message (CTS)	Matches the message sent to other clients when a user changes status to 'away'.	AIM Away Message
aim-buddy-comment (ANY)	Matches the comment stored for a buddy in the contact list.	AIM Buddy Comment
aim-capabilities (ANY)	Matches the set of features supported by the client.	AIM Capabilities
aim-chat-info (STC)	Matches the information about a chatroom.	AIM Chat Info
aim-chat-interests (STC)	Matches the categories of personal interests in a user's profile.	AIM Chat Interests
aim-chat-room-desc (STC)	Matches the description of a chatroom.	AIM Chat Room Desc
aim-chat-room-name (STC)	Matches the name of a chatroom in an AIM/ICQ session.	AIM Chat Room Name
aim-client-ip (STC)	Matches the IP address of the client for direct P2P communication.	AIM Client Ip
aim-client-port (STC)	Matches the port that the client is listening on for P2P communication.	AIM Client Port
aim-client-status (STC)	Matches the user's online status.	AIM Client Status
aim-decline-reason (ANY)	Matches the decline reason when a client refuses to be added to another user's contact list.	AIM Decline Reason
aim-descripted-url (ANY)	Matches the description and URL when sending a Web page to another address.	AIM Descripted Url
aim-email-address (STC)	Matches the e-mail address of a user as it appears in the profile.	AIM Email Address
aim-error-url (STC)	Matches the URL on the server where the user can reconfigure the account password.	AIM Error Url
aim-gcard-message (ANY)	Matches the message associated with a greeting card.	AIM Gcard Message

Table 27: Service Contexts: AIM (continued)

Context and Direction	Description	Display Name
aim-gcard-recipient (ANY)	Matches the screen name of a greeting card recipient.	AIM Gcard Recipient
aim-gcard-sender (ANY)	Matches the screen name of a greeting card sender.	AIM Gcard Sender
aim-gcard-theme (ANY)	Matches the theme of a greeting card sent from one client to another.	AIM Gcard Theme
aim-gcard-title (ANY)	Matches the title of a greeting card sent from one user to another.	AIM Gcard Title
aim-gcard-url (ANY)	Matches the URL of the greeting card sent from one user to another.	AIM Gcard Url
aim-get-file (STC)	Matches the name of a file that the user is transferring from a peer.	AIM Get File
aim-group (ANY)	Matches the name of a group of items (usually buddies).	AIM Group
aim-info-text (STC)	Matches additional information text that appears in a user's profile.	AIM Info Text
aim-local-ip (CTS)	Matches the IP address of a client used for P2P communication.	AIM Local Ip
aim-local-port (CTS)	Matches the local port that the client is listening on for P2P communication.	AIM Local Port
aim-message-block (ANY)	Matches the instant message sent from one user to another.	AIM Message Block
aim-message-description (ANY)	Matches the description of a message.	AIM Message Description
aim-nick-name (ANY)	Matches the nickname of an AIM/ICQ user.	AIM Nick Name
aim-oft-content (ANY)	Matches the contents of a file being transferred between peers.	AIM Oft Content
aim-oft-name (ANY)	Matches the name of a file being transferred between peers.	AIM Oft Name
aim-peer-ip (STC)	Matches the IP address of a peer for direct P2P communication.	AIM Peer Ip
aim-peer-port (STC)	Matches the port of a peer for direct P2P communication.	AIM Peer Port

Table 27: Service Contexts: AIM (continued)

Context and Direction	Description	Display Name
aim-put-file (CTS)	Matches the name of a file that the user is transferring to a peer.	AIM Put File
aim-screen-name (ANY)	Matches the screen name of a user.	AIM Screen Name
aim-server-ip (STC)	Matches the IP address of a server. Typically used when the main server redirects the client to another server.	AIM Server Ip
aim-server-url (STC)	Matches any URL on the server.	AIM Server Url
aim-url (ANY)	Matches the URL of a user's profile.	AIM Url
aim-xml-value (STC)	Matches the XML string sent by the server with the value of a requested URL.	AIM Xml Value

Table 28: Service Contexts: BGP

Context and Direction	Description	Display Name
bgp-keepalive-msg (ANY)	Matches the BGP keep alive message.	BGP KeepAlive Message
bgp-message (ANY)	Matches any BGP message.	BGP Message
bgp-notification-msg (ANY)	Matches the BGP notification message.	BGP Notification Message
bgp-open-msg (ANY)	Matches the BFP open message.	BGP Open Message
bgp-open-no-parm (ANY)	Matches the BFP open message without optional parameters.	BGP Open Message without optional parameters
bgp-open-parm (ANY)	Matches the optional parameters in the BGP open message.	BGP Optional parameters in Open Message
bgp-route-refresh-msg (ANY)	Matches the BGP Route Refresh Message	BGP Route Refresh Message
bgp-update-attr-aggregator (ANY)	Matches the Aggregator path attribute data in the BGP update message.	BGP Aggregator Path Attribute in Update Message
bgp-update-attr-as-path (ANY)	Matches the AS path attribute data in the BGP update message.	BGP AS-Path Path Attribute in Update Message
bgp-update-attr-atomic-aggr (ANY)	Matches the atomic-aggregator path attribute data in the BGP update message.	BGP Atomic-Aggregator Path Attribute in Update Message

Table 28: Service Contexts: BGP (continued)

Context and Direction	Description	Display Name
bgp-update-attr-cluster-list (ANY)	Matches the Cluster-List path attribute data in the BGP update message.	BGP Cluster-List Path Attribute in Update Message
bgp-update-attr-communities (ANY)	Matches the Communities path attribute data in the BGP update message.	BGP Communities Path Attribute in Update Message
bgp-update-attr-local-pref (ANY)	Matches the Local-Pref path attribute data in BGP update message.	BGP Local-Pref Path Attribute in Update Message
bgp-update-attr-med (ANY)	Matches the Multi-Exit-Disc path attribute data in the BGP update message.	BGP Multi-Exit-Disc Path Attribute in Update Message
bgp-update-attr-next-hop (ANY)	Matches the Next-Hop path attribute data in the BGP update message.	BGP Next-Hop Path Attribute in Update Message
bgp-update-attr-nonstd (ANY)	Matches any Non-Standard path attribute data in the BGP update message.	BGP Non-standard Path Attribute in Update Message
bgp-update-attr-rigin (ANY)	Matches the Origin path attribute date in the BGP update message.	BGP Origin Path Attribute in Update Message
bgp-updet-attr-originator (ANY)	Matches the Originator path attribute data in BFP update message.	BGP Originator Path Attribute in Update Message
bgp-update-msg (ANY)	Matches the BGP update message.	BGP Update Message
bgp-update-nlri_infor (ANY)	Matches the Network Layer Reachability Information in the BGP update message.	BGP Network Layer Reachability Information in Update Message
bgp-update-norm-unfeasible-rte (ANY)	Matches the unfeasible routes data in BFP update message. This context shows each route expanded to 4 bytes, prefixed by a delimiter.	BGP Unfeasible routes in Update Message (Normalized)
bgp-update-total-path-attribute (ANY)	Matches the Total Path Attribute data in the BGP update message.	BGP Total Path Attributes in Update Message
bgp-update-unfeasible-rts (ANY)	Matches the unfeasible routes data in the BGP update message.	BGP Unfeasible routes in Update Message

Table 29: Service Contexts: DHCP

Context and Direction	Description	Display Name
dhcp-file-name (ANY)	Matches the filename in a DHCP/bootp message.	DHCP File Name
dhcp-option (ANY)	Matches each option in a DHCP/bootp message. Each option context contains the type and length of the option.	DHCP Option
dhcp-server-name (ANY)	Matches the server name in a DHCP/bootp message.	DHCP Server Name

Table 30: Service Contexts: DNS

Context and Direction	Description	Display Name
dns-cname (ANY)	Matches the CNAME in a DNS request or response.	DNS Cname
dns-rr-a6-rdata (ANY)	Match the rdata of an A6 RR in a DNS request response.	DNS A6 Rdata
dns-rr-afsdb-rdata (ANY)	Matches the rdata of an AFSDB RR in a DNS request or response.	DNS AFSDB Rdata
dns-rr-apl-rdata (ANY)	Matches the rdata of an APL RR in a DNS request or response.	DNS APL Rdata
dns-rr-atma-rdata (ANY)	Matches the rdata of an ATMA RR in a DNS request or response.	DNS ATMA Rdata
dns-rr-cname-rdata (ANY)	Matches the rdata of a CNAME RR in a DNS request or response.	DNS CNAME Rdata
dns-rr-dnskey-rdata (ANY)	Matches the rdata of DNSKEY RR in a DNS request or response.	DNS DNSKEY Rdata
dns-rr-ds-rdata (ANY)	Matches the rdata of a DN RR in a DNS request or response.	DNS DS Rdata
dns-rr-eid-rdata (ANY)	Matches the rdata of an EID RR in a DNS request or response.	DNS EID Rdata
dns-rr-hinfo-rdata (ANY)	Matches the rdata of an HINFO RR in a DNS request or response.	DNS HINFO Rdata
dns-rr-key-rdata (ANY)	Matches the rdata of a KEY RR in a DNS request or response.	DNS KEY Rdata
dns-rr-kx-rdata (ANY)	Matches the rdata of a KX RR in a DNS request or response.	DNS KX Rdata

Table 30: Service Contexts: DNS (continued)

Context and Direction	Description	Display Name
dns-rr-mb-rdata (ANY)	Matches the rdata of an MB RR in a DNS request or response.	DNS MB Rdata
dns-rr-md-rdata (ANY)	Matches the rdata of an MD RR in a DNS request or response.	DNS MD Rdata
dns-rr-mf-rdata (ANY)	Matches the rdata of an MF RR in a DNS request or response.	DNS MF Rdata
dns-rr-mg-rdata (ANY)	Matches the rdata of an MG RR in a DNS request or response.	DNS MG Rdata
dns-rr-minfo-rdata (ANY)	Matches the rdata of an MINFO RR in a DNS request or response.	DNS MINFO Rdata
dns-rr-mr-rdata (ANY)	Matches the rdata of an MR RR in a DNS request or response.	DNS MR Rdata
dns-rr-mx-rdata (ANY)	Matches the rdata of an MX RR in a DNS request or response.	DNS MX Rdata
dns-rr-naptr-rdata (ANY)	Matches the rdata of a NAPTR RR in a DNS request or response.	DNS NAPTR Rdata
dns-rr-nimloc-rdata (ANY)	Matches the rdata of an NIMLOC RR in a DNS request or response.	DNS NIMLOC Rdata
dns-rr-ns-rdata (ANY)	Matches the rdata of an NS RR in a DNS request or response.	DNS NSAPPTR Rdata
dns-rr-nsap-rdata (ANY)	Matches the rdata of an NSAP RR in a DNS request or response.	DNS NSAP Rdata
dns-rr-ns-rdata (ANY)	Matches the rdata of an NS RR in a DNS request or response.	DNS NSEC Rdata
dns-rr-nsapptr-rdata (ANY)	Matches the rdata of an NSAPPTR RR in a DNS request or response.	DNS NS Rdata
dns-rr-nsec-rdata (ANY)	Matches the rdata of an NSEC RR in a DNS request or response.	DNS NULL Rdata
dns-rr-null-rdata (ANY)	Matches the rdata of a NULL RR in a DNS request or response.	DNS NXT Rdata
dns-rr-nxt-rdata (ANY)	Matches the rdata of a NXT RR in a DNS request or response.	DNS OPT Rdata

Table 30: Service Contexts: DNS (continued)

Context and Direction	Description	Display Name
dns-rr-ptr-rdata (ANY)	Matches the rdata of a PTR RR in a DNS request or response.	DNS PTR Rdata
dns-rr-px-rdata (ANY)	Matches the rdata of a PX RR in a DNS request or response.	DNS PX Rdata
dns-rr-rp-rdata (ANY)	Matches the rdata of an RP RR in a DNS request or response.	DNS RP Rdata
dns-rr-rrsig-rdata (ANY)	Matches the rdata of an RRSIG RR in a DNS request or response.	DNS RRSIG Rdata
dns-rr-sig-rdata (ANY)	Matches the rdata of an SIG RR in a DNS request or response	DNS SIG Rdata
dns-rr-soa-rdata (ANY)	Matches the rdata of an SOA RR in a DNS request or response.	DNS SOA Rdata
dns-rr-sshfp-data (ANY)	Matches the rdata of an SSHFP RR in a DNS request or response.	DNS SSHFP Rdata
dns-rr-tsip-rdata (ANY)	Matches the rdata of a TSIP RR in a DNS request or response.	DNS TSIP Rdata
dns-rr-txt-rdata (ANY)	Matches the rdata of a TXT RR in a DNS request or response.	DNS TXT Rdata
dns-rr-type-rdata (ANY)	Matches the entire resource record in a DNS request or response, including the type and class.	DNS Type RData
dns-rr-wks-rdata (ANY)	Matches the rdata of a WKS RR in a DNS request or response.	DNS WKS Rdata
dns-type-name (ANY)	Matches any name resource record in a DNS request or response. The first 2 bytes of the context contain the RFC-1035 type values.	DNS Type Name
dns-update-header	Matches the header of a DNS UPDATE request or response.	DNS UPDATE header

Table 31: Service Contexts: Finger

Context and Direction	Description	Display Name
finger-host (CTS)	Matches each hostname in a FINGER request.	FINGER Host

Table 31: Service Contexts: Finger (continued)

Context and Direction	Description	Display Name
finger-s2c-data (STC)	finger-s2c-data	finger-s2c-data
finger-user (CTS)	Matches the username in a FINGER request.	FINGER User

Table 32: Service Contexts: First Data Packet

Context and Direction	Description	Display Name
first-data-packet (ANY)	Matches the first data packet of a session.	First Data Packet
first-packet (ANY)	Matches the first packet of a session.	First Packet

Table 33: Service Contexts: FTP

Context and Direction	Description	Display Name
ftp-account (CTS)	Matches the FTP login account name.	FTP Account
ftp-banner (STC)	Matches the banner returned by the server at the start of an FTP session.	FTP Banner
ftp-command (CTS)	Matches each of the FTP command names.	FTP Command
ftp-cwd-pathname (CTS)	Matches the directory name in the CWD command of an FTP session.	FTP Cwd Pathname
ftp-dele-pathname (CTS)	Matches the file name in the DELE command of an FTP session.	FTP Dele Pathname
ftp-get-filename (CTS)	Matches the filename in the GET command of an FTP session.	FTP Get Filename
ftp-list-pathname (CTS)	Matches the directory or file name in the LIST command of an FTP session.	FTP List Pathname
ftp-mkd-pathname (CTS)	Matches the directory name in the MKD command of an FTP session.	FTP Mkd Pathname
ftp-nlst-pathname (CTS)	Matches the directory or file name in the NLST command of an FTP session.	FTP Nlst Pathname

Table 33: Service Contexts: FTP (continued)

Context and Direction	Description	Display Name
ftp-password (CTS)	Matches the FTP login password.	FTP Password
ftp-pathname (CTS)	Matches a directory or file name in any of the FTP commands.	FTP Pathname
ftp-put-filename (CTS)	Matches the filename in the PUT command of an FTP session.	FTP Put Filename
ftp-reply-100-line (STC)	Matches the FTP 1yz Positive Preliminary reply.	FTP Reply 100 Line
ftp-reply-200-line (STC)	Matches the FTP 2yz Positive Completion reply.	FTP Reply 200 Line
ftp-reply-300-line (STC)	Matches the FTP 3yz Positive Intermediate reply.	FTP Reply 300 Line
ftp-reply-400-line (STC)	Matches the FTP 4yz Transient Negative Completion reply.	FTP Reply 400 Line
ftp-reply-500-line (STC)	Matches the FTP 5yz Permanent Negative Completion reply.	FTP Reply 500 Line
ftp-reply-line (STC)	Matches the FTP reply line.	FTP Reply Line
ftp-request (CTS)	Matches FTP request line (command and arguments).	FTP Request
ftp-rmd-pathname (CTS)	Matches the directory name in the RMD command of an FTP session.	FTP Rmd Pathname
ftp-rnfr-pathname (CTS)	Matches a directory or file name in the RNFR command of an FTP session.	FTP Rnfr Pathname
ftp-rnto-pathname (CTS)	Matches a directory or file name in the RNTO command of an FTP session.	FTP Rnto Pathname
ftp-sitestring (CTS)	Matches the arguments of the SITE command in an FTP session.	FTP Sitestring
ftp-smnt-pathname (CTS)	Matches the directory or file name in the SMNT command of an FTP session.	FTP Smnt Pathname
ftp-stat-pathname (CTS)	Matches the directory or file name in the STAT command of an FTP session.	FTP Stat Pathname

Table 33: Service Contexts: FTP (continued)

Context and Direction	Description	Display Name
ftp-username (CTS)	Matches the FTP login user name.	FTP Username

Table 34: Service Contexts: Gnutella

Context and Direction	Description	Display Name
gnutella-connect-fail-reason (STC)	Matches the connection fail reason string in a Gnutella connection.	GNUTELLA Connect Fail Reason
gnutella-connect-header (ANY)	Matches the contents of the HTTP style CONNECT message in a Gnutella session.	GNUTELLA Connect Header
gnutella-http-get-filename (CTS)	Matches the name of the file that the client intends to retrieve.	GNUTELLA Http Get Filename
gnutella-http-header (ANY)	Matches any HTTP style headers in a Gnutella session.	GNUTELLA Http Header
gnutella-queryhit-vendor (STC)	Matches the 4-byte vendor code in the reply for the QUERYHIT message.	GNUTELLA Queryhit Vendor
gnutella-search-criteria (CTS)	Matches the search criteria in a QUERY message of a Gnutella session.	GNUTELLA Search Criteria
gnutella-user-agent (ANY)	Matches the name of the user agent in a Gnutella session.	GNUTELLA User Agent

Table 35: Service Contexts: Gopher

Context and Direction	Description	Display Name
gopher-display (STC)	Matches the display string of a Gopher item.	GOPHER Display
gopher-file (STC)	Matches the contents of a Gopher item/file.	GOPHER File
gopher-host-port (STC)	Matches the host and port used to get an item.	GOPHER Host Port
gopher-selector (STC)	Matches the selector string of a Gopher item.	GOPHER Selector

Table 36: Service Contexts: H225

Context and Direction	Description	Display Name
h225ras-admission (ANY)	Matches H225RAS admission messages (AdmissionConfirm, AdmissionReject, AdmisssonRequest).	H225RAS Admission
h225ras-bandwidth (ANY)	Matches H225RAS bandwidth messages (BandwidthConfirm, BandwidthReject, BandwidthRequest).	H225RAS Bandwidth
h225ras-command-state (ANY)	Matches the state of the H225RSA connection.	H225RAS Command State
h225ras-disengage (ANY)	Matches H225RAS disengage messages (DisengageConfirm, DisengageReject, DisengageRequest).	H225RAS Disengage
h225ras-gatekeeper (ANY)	Matches H225RAS gatekeeper messages (GatekeeperConfirm, GatekeeperReject, GatekeeperRequest).	H225RAS Gatekeeper
h225ras-info (ANY)	Matches H225RAS informational messages (InfoRequestAck, InfoRequestResponse, InfoRequest).	H225RAS Info
h225ras-location (ANY)	Matches H225RAS location messages (LocationConfirm, LocationReject, LocationRequest).	H225RAS Location
h225ras-message (ANY)	Matches the broad H225RAS message context.	H225RAS Message
h225ras-nonstandard (ANY)	Matches the H225RAS nonstandard message context.	H225RAS Non Standard
h225ras-registration (ANY)	Matches the H225RAS registration message.	H225RAS Registration
h225ras-resource (ANY)	Matches H225RAS resources available messages (ResourcesAvailableConfirm, ResourcesAvailableIndicate).	H225RAS Resource
h225ras-rip (STC)	Matches the H225RAS request-in-progress message.	H225RAS Request in Progress
h225ras-servicecontrol (CTS)	Matches the H225RAS service control message.	H225RAS ServiceControl

Table 36: Service Contexts: H225 (continued)

Context and Direction	Description	Display Name
h225ras-unknown-message (ANY)	Match the H225RAS Unknown message type.	H225RAS Unknown Message Type
h225ras-unregistration (ANY)	Matches the H225RAS unregistration message.	H225RAS Unregistration
h225ras-unspecified-message (ANY)	Matches the H225RAS unspecified message.	H225RAS Unspecified Message
h225ras-version (ANY)	Matches the H225RAS version message.	H225RAS Version
h225sgn-message (ANY)	Matches the H225SGN message body started with the message-type byte.	H225SGN Message
h225sgn-preamble (ANY)	Matches the H225SGN signaling protocol discriminator and call reference value.	H225 Signaling Protocol Preamble

Table 37: Service Contexts: HTTP

Context and Direction	Description	Display Name
http-authorization (CTS)	Matches the username and password decoded from the Authorization: Basic header in an HTTP request.	HTTP Authorization
http-data (ANY)	Matches any HTTP data in an HTTP transaction that is not text/html, text/plain, or FORM values in a POST request.	HTTP Data
http-first-data-chunk (ANY)	Matches the first data chunk in an HTTP transaction.	HTTP FIRST DATA CHUNK
http-flash		HTTP Flash
http-form-data (CTS)	Matches each of the form values in a POST request of an HTTP transaction.	HTTP Form Data
http-get-url (CTS)	Matches the URL in an HTTP get request as it appears in the stream.	HTTP GET URL
http-get-url-parsed (CTS)	Matches the decoded, normalized URL in an HTTP get request.	HTTP GET URL Parsed

Table 37: Service Contexts: HTTP (continued)

Context and Direction	Description	Display Name
http-get-url-parsed-param (CTS)	Matches the decoded, normalized URL in an HTTP get request along with any CGI parameters.	HTTP GET URL Parsed Param
http-get-url-parsed-param-parsed (CTS)	Matches the decoded, normalized URL in and HTTP GET request along with the any decoded CGI parameters.	HTTP GET URL Parsed Param Parsed
http-head-url (CTS)	Matches the URL in an HTTP head request as it appears in the stream.	HTTP HEAD URL
http-head-url-parsed (CTS)	Matches the decoded, normalized URL in an HTTP head request.	HTTP HEAD URL Parsed
http-header (ANY)	Matches any HTTP header.	HTTP Header
http-header-accept (CTS)	Matches each Accept: header in an HTTP request.	HTTP Header Accept
http-header-accept-encoding (CTS)	Matches each Accept-Encoding: header in an HTTP request.	HTTP Header Accept Encoding
http-header-accept-language (CTS)	Matches each Accept-Language: header in an HTTP request.	HTTP Header Accept Language
http-header-content-encoding (ANY)	Matches each Content-Encoding: header in an HTTP transaction.	HTTP Header Content Encoding
http-header-content-language (ANY)	Matches each Content-Language: header in an HTTP transaction.	HTTP Header Content Language
http-header-content-location (ANY)	Matches each Content-Location: header in an HTTP transaction.	HTTP Header Content Location
http-header-content-md5 (ANY)	Matches each Content-MD5: header in an HTTP transaction.	HTTP Header Content Md5
http-header-content-type (ANY)	Matches each Content-Type: header in an HTTP transaction.	HTTP Header Content Type
http-header-cookie (ANY)	Matches each Cookie: header in an HTTP transaction.	HTTP Header Cookie
http-header-host (CTS)	Matches each Host: header in an HTTP request.	HTTP Header Host
http-header-referer (CTS)	Matches each Referrer: header in an HTTP request.	HTTP Header Referer

Table 37: Service Contexts: HTTP (continued)

Context and Direction	Description	Display Name
http-header-server (STC)	Matches each Server: header in an HTTP reply.	HTTP Header Server
http-header-soapaction (ANY)	Matches each soapaction: header in an HTTP transaction.	HTTP Soap Action
http-header-user-agent (CTS)	Matches each User-Agent: header in an HTTP request.	HTTP Header User Agent
http-image (ANY)	Matches IMATE contents (BMP, PNG) in HTTP transaction.	HTTP IMAGE
http-jpeg-raw (ANY)	Matches JPEG content in HTTP transaction.	HTTP JPEG RAW
http-jpeg-tag (ANY)	Matches JPEG tag of JPEG content in HTTP transaction.	HTTP JPEG TAG
http-object-tag-clsid (STC)	Matches the CLSID of an object tag.	HTTP Object Tag CLSID
http-param-parsed (CTS)	Matches the decoded CGI parameters in an HTTP request.	HTTP Param Parsed
http-pdf	HTTP PDF	HTTP PDF
http-png-chunk (ANY)	Matches contents of PNG chunk to HTTP transaction.	HTTP PNG CHUNK
http-post-url (CTS)	Matches the URL in an HTTP post request as it appears in the stream.	HTTP POST URL
http-post-url-parsed (CTS)	Matches the decoded, normalized URL in an HTTP post request.	HTTP POST URL Parsed Param Parsed
http-post-variable (CTS)	Matches each CGI variable in the form data of an HTTP POST request.	HTTP POST Variable
http-post-variable-parsed (CTS)	Matches each decoded CGI variable in the form data of an HTTP POST request.	HTTP POST Variable Parsed
http-request (CTS)	Matches each HTTP request line.	HTTP Request
http-request-method (CTS)	Matches the method name in an HTTP request.	HTTP Request Method
http-status (STC)	Matches the status line in an HTTP reply.	HTTP Text Html

Table 37: Service Contexts: HTTP (continued)

Context and Direction	Description	Display Name
http-text-html (ANY)	Matches the text/html data in an HTTP transaction.	HTTP Text Html
http-text-html-body (ANY)	Matches the body of text/html data in an HTTP tranaction	HTTP Text Html body
http-text-html-head (ANY)	Matches the header of text/html data in an HTTP transaction.	HTTP Text Html header
http-text-html-script (ANY)	Matches the script tag of text/html data in an HTTP transaction.	HTTP Text Html script
http-text-html-style (ANY)	Matches the style tag of text/html data in an HTTP transaction.	HTTP Text Html style
http-text-html-tag (ANY)	Matches any tag inside text/html data in an HTTP transaction.	HTTP Text Html Tag
http-text-plain (ANY)	Matches the text/plain data in an HTTP transaction.	HTTP Text Plain
http-text-soap (ANY)	Matches the text/soap data in and HTTP transaction.	HTTP Text SOAP
http-text-xml (ANY)	Matches the tex/xml data in an HTTP transaction.	HTTP Text Xml
http-url (CTS)	Matches the URL in an HTTP request as it appears in the stream.	HTTP URL
http-url-parsed (CTS)	Matches the decoded, normalized URL in an HTTP request.	HTTP URL Parsed
http-url-parsed-param (CTS)	Matches the decoded, normalized URL in an HTTP request along with the CGI parameters, if any	HTTP URL Parsed Param
http-url-parsed-param-parsed (CTS)	Matches the decoded, normalized URL in an HTTP request along with the decoded CGI parameters, if any	HTTP URL Parsed Param Parsed
http-url-variable (CTS)	Matches each CGI variable in the URL of an HTTP GET request.	HTTP URL Variable
http-url-variable-parsed (CTS)	Matches each decoded CGI variable in the URL of an HTTP GET request.	HTTP URL Variable Parsed
http-variable (CTS)	Matches each CGI variable in an HTTP GET or POST request.	HTTP Variable

Table 37: Service Contexts: HTTP (continued)

Context and Direction	Description	Display Name
http-variable-parsed (CTS)	Matches each decoded CGI variable in an HTTP GET or POST request.	HTTP Variable Parsed

Table 38: Service Contexts: IEC

Context and Direction	Description	Display Name
iec104-message-type-i (ANY)	Matches the Type-I message of IEC104.	IEC104 Message Type I
iec104-message-type-s (ANY)	Matches the Type-S message of IEC104.	IEC104 Message Type S
iec104-message-type-u (ANY)	Matches the Type-U message of IEC104.	IEC104 Message Type U

Table 39: Service Contexts: IKE

Context and Direction	Description	Display Name
ike-payload (ANY)	Matches the payload in an IKE transaction	IKE payload

Table 40: Service Contexts: IMAP

Context and Direction	Description	Display Name
imap-append (CTS)	Matches the e-mail contents in an IMAP append message.	IMAP Append
imap-append-line (CTS)	Matches arguments of IMAP Append command line in an IMAP session.	IMAP Append Argument
imap-authenticate (CTS)	Matches arguments of IMAP Authenticate command in an IMAP session.	IMAP Authenticate
imap-banner-(STC)	Matches arguments of the fist untagged OK response from an IMAP session.	IMAP BANNER
imap-command (CTS)	Matches each IMAP command name in an IMAP session.	IMAP Command
imap-command-line (CTS)	Matches each IMAP command name and arguments in an IMAP session.	IMAP Command Line
imap-copy (CTS)	Matches arguments of IMAP Copy command in an IMAP session.	IMAP Copy

Table 40: Service Contexts: IMAP (continued)

Context and Direction	Description	Display Name
imap-create (CTS)	Matches arguments of IMAP Create command in an IMAP session.	IMAP Create
imap-delete (CTS)	Matches arguments of IMAP Delete command in an IMAP session.	IMAP Delete
imap-deleteacl (CTS)	Matches arguments of IMAP DeleteACL command in an IMAP session.	IMAP DeleteACL
imap-examine (CTS)	Matches arguments of IMAP Examine command in an IMAP session.	IMAP Examine
imap-fetch (CTS)	Matches arguments of IMAP Fetch command in an IMAP session.	IMAP Fetch
imap-getacl (CTS)	Matches arguments of IMAP GetACL command in an IMAP session.	IMAP GetACL
imap-list (CTS)	Matches arguments of IMAP List/RList command in an IMAP session.	IMAP List
imap-listrights (CTS)	Matches arguments of IMAP ListRights command in an IMAP session.	IMAP ListRights
imap-login (CTS)	Matches arguments of IMAP Login command in an IMAP session.	IMAP Login
imap-lsub (CTS)	Matches arguments of IMAP LSUB/RLSUB command in an IMAP session.	IMAP LSUB
imap-mailbox (CTS)	Matches each mailbox name in an IMAP session.	IMAP Mailbox
imap-myrights (CTS)	Matches arguments of IMAP MyRights command in an IMAP session.	IMAP MyRights
imap-rename (CTS)	Matches arguments of IMAP Rename command in an IMAP session.	IMAP Rename
imap-search (CTS)	Matches arguments of IMAP Search command in an IMAP session.	IMAP Search
imap-select (CTS)	Matches arguments of IMAP Select command in an IMAP session.	IMAP Select
imap-setacl (CTS)	Matches arguments of IMAP SetACL command in an IMAP session.	IMAP SetACL

Table 40: Service Contexts: IMAP (continued)

Context and Direction	Description	Display Name
imap-status (CTS)	Matches arguments of IMAP Status command in an IMAP session.	IMAP Status
imap-store (CTS)	Matches arguments of IMAP Store command in an IMAP session.	IMAP Store
imap-subscribe (CTS)	Matches arguments of IMAP Subscribe command in an IMAP session.	IMAP Subscribe
imap-uid (CTS)	Matches arguments of IMAP UID command in an IMAP session.	IMAP UID
imap-unsubscribe (CTS)	Matches arguments of IMAP Unsubscribe command in an IMAP session.	IMAP Unsubscribe
imap-user (CTS)	Matches the IMAP user name in an IMAP session.	IMAP User

Table 41: Service Contexts: IRC

Context and Direction	Description	Display Name
irc-command (ANY)	Matches any IRC command name.	IRC Command
irc-join-chan (ANY)	Matches the channel name in the JOIN command of an IRC session.	IRC Join Chan
irc-nick-name (ANY)	Matches the name in the NICK command of an IRC session.	IRC Nick Name
irc-notice-msg (ANY)	Matches the message in the NOTICE command of an IRC session.	IRC Notice Msg
irc-oper-name (ANY)	Matches the name in the OPER command of an IRC session.	IRC Oper Name
irc-oper-password (ANY)	Matches the password in the OPER command of an IRC session.	IRC Oper Password
irc-part-chan (ANY)	Matches the channel name in the PART command of an IRC session.	IRC Part Chan
irc-password (ANY)	Matches the password in the PASS command of an IRC session.	IRC PasswordIRC Priv Msg
irc-priv-msg (ANY)	Matches the message in the PRIVMSG command of an IRC session.	IRC Priv Msg

Table 41: Service Contexts: IRC (continued)

Context and Direction	Description	Display Name
irc-real-name (ANY)	Matches the real name in the USER command of an IRC session.	IRC Real Name
irc-topic (ANY)	Matches the arguments of the TOPIC command of an IRC session.	IRC Topic
irc-user-name (ANY)	Matches the name in the USER command of an IRC session.	IRC User Name

Table 42: Service Contexts: LDAP

Context and Direction	Description	Display Name
ldap-abandon-request (CTS)	Matches the entire Abandon Request message.	LDAP Abandon Request
ldap-add-request (CTS)	Matches the entire Add Request message.	LDAP Add Request
ldap-add-request-attribute (CTS)	Matches each attribute in an Add Request message. The values are NULL delimited and the type, and values are newline delimited.	LDAP Add Request Attribute
ldap-add-request-attributetype (CTS)	Matches the type each attribute in an Add Request message.	LDAP Add Request Attribute Type
ldap-add-request-attributevalue (CTS)	Matches the value of each attribute in an Add Request message.	LDAP Add Request Attribute Value
ldap-add-request-entry (CTS)	Matches the object in an Add Request message.	LDAP Add Request Entry
ldap-bind-request (CTS)	Matches the entire LDAP Bind Request message.	LDAP Bind Request
ldap-bind-request-authentication (CTS)	Matches the authentication information in a Bind Request message including the 1-byte type.	LDAP Bind Request Authentication
ldap-bind-request-ldapDN (CTS)	Matches the name of the directory object to which the client wants to bind.	LDAP Bind Request LdapDN
ldap-bind-request-version (CTS)	Matches the LDAP version in a Bind Request message.	LDAP Bind Request Version
ldap-compare-request (CTS)	Matches the entire Compare Request message.	LDAP Compare Request

Table 42: Service Contexts: LDAP (continued)

Context and Direction	Description	Display Name
ldap-compare-request-assertionvalue (CTS)	Matches the value against which the attribute value is compared in a Compare Request message.	LDAP Compare Request Assertion Value
ldap-compare-request-attributedesc (CTS)	Matches the attribute type of an entry in a Compare Request message.	LDAP Compare Request Attribute Type
ldap-compare-request-entry (CTS)	Matches the entry of the DN to be compared in a Compare Request message.	LDAP Compare Request Entry
ldap-delete-request (CTS)	Matches the entire Delete Request message.	LDAP Data
ldap-extended-request (CTS)	Matches the entire Extended Request message.	LDAP Delete Request
ldap-extended-request-requestName (CTS)	Matches the request name in the Extended Request message.	LDAP Extended Request
ldap-extended-request-requestValue (CTS)	Matches the request value in the Extended Request message.	LDAP Extended Request Name
ldap-extended-response-response (STC)	Matches the response field in the Extended Request message.	LDAP Extended Request Value
ldap-extended-response-responseName (STC)	Matches the response name in the Extended Response message.	LDAP Extended Response Response
ldap-modify-request (CTS)	Matches the entire Modify Request message.	LDAP Extended Response Name
ldap-modify-request-attribute (CTS)	Matches each attribute in a Modify Request message including the 1-byte modify operation. The values are NULL delimited, and the type and values are newline delimited.	LDAP ModifyDN Request
ldap-modify-request-attributetype (CTS)	Matches each attribute type in a Modify Request message.	LDAP ModifyDN Request Entry
ldap-modify-request-attributevalue (CTS)	Matches each attribute value in a Modify Request message.	LDAP ModifyDN Request NewRDN
ldap-modify-request-object (CTS)	Matches the object in the Modify Request message.	LDAP ModifyDN Request Newsuperior
ldap-modifyDN-request (CTS)	Matches the entire Modify-DN Request message.	LDAP Modify Request

Table 42: Service Contexts: LDAP (continued)

Context and Direction	Description	Display Name
ldap-modifyDN-request-entry (CTS)	Matches the DN of the entry in a Modify-DN Request message.	LDAP Modify Request Attribute
ldap-modifyDN-request-newRDN (CTS)	Matches the new DN that replaces the old DN in a Modify-DN Request message.	LDAP Modify Request Attribute Type
ldap-modifyDN-request-newsuperior (CTS)	Matches the new DN that becomes the parent of the existing DN entry in a Modify-DN Request message.	LDAP Modify Request Attribute Value
ldap-result (STC)	Matches the entire Result message, including the 1-byte response type.	LDAP Modify Request Object
ldap-result-errorMessage (STC)	Matches the error message in the result.	LDAP Result
ldap-result-matchedDN (STC)	Matches the base object in the Result message, including the 1-byte tag.	LDAP Result ErrorMessage
ldap-result-referral (STC)	Matches each referral URL in the result.	LDAP Result MatchedDN
ldap-search-request (CTS)	Matches the entire LDAP Search Request message.	LDAP Result Referral
ldap-search-request-attribute (CTS)	Matches each attribute in a Search Request message.	LDAP Search Request
ldap-search-request-attributelist (CTS)	Matches all the attributes in a Search Request message.	LDAP Search Request Attribute
ldap-search-request-baseObject (CTS)	Matches the base object entry against which the search is performed. This includes the 1-byte scope, which can represent baseObject, singleLevel or wholeSubtree.	LDAP Search Request Attributelist
ldap-search-request-filter (CTS)	Matches the contents of the search filter.	LDAP Search Request BaseObject
ldap-search-request-sizeLimit (CTS)	Matches the sizeLimit field of the search request.	LDAP Search Request Filter
ldap-search-request-timeLimit (CTS)	Matches the timeLimit field of the search request.	LDAP Search Request SizeLimit
ldap-search-resentry (STC)	Matches the entire Search Result message.	LDAP Search Request TimeLimit

Table 42: Service Contexts: LDAP (continued)

Context and Direction	Description	Display Name
ldap-search-resentry-attribute (STC)	Matches each attribute in the search result. The values are NULL delimited, and the type and value list are newline delimited.	LDAP Search Resentry
ldap-search-resentry-attributetype (STC)	Matches each attribute type in the search result.	LDAP Search Resentry Attribute
ldap-search-resentry-attributevalue (STC)	Matches each attribute value in the search result.	LDAP Search Resentry Attributetype
ldap-search-resentry-objectname (STC)	Matches the base object of the search result.	LDAP Search Resentry Objectname
ldap-search-resref (STC)	Matches the entire Search Result Reference message.	LDAP Search Resref
ldap-search-resref-referral (STC)	Matches each referral URL in the Search Result Reference message.	LDAP Search Resref Referral

Table 43: Service Contexts: Line

Context and Direction	Description	Display Name
line (ANY)	Matches a line extracted from the reassembled, normalized TCP stream data. This context is available for only those protocols that are line based.	Line

Table 44: Service Contexts: LPR

Context and Direction	Description	Display Name
lpr-cfile-command (CTS)	Matches the entire CFILE subcommand line, including the first byte of the subcommand type.	LPR Cfile Command
lpr-cfile-name (CTS)	Matches the name of the control filename that is sent as part of the RECEIVE-JOB command.	LPR Cfile Name
lpr-command (CTS)	Matches the entire command line, including the first byte of the command code.	LPR Command
lpr-dfile-name (CTS)	Matches the name of the data filename that is sent as part of the RECEIVE-JOB command.	LPR Dfile Name

Table 45: Service Contexts: MGCP

Context and Direction	Description	Display Name
mgcp-call-id (ANY)	Matches the MGCP call ID parameter value.	MGCP Call ID
mgcp-command (ANY)	Matches the MGCP command line.	MGCP Command
mgcp-ep-name (ANY)	Matches the MGCP endpoint name specified in command line or command parameters.	MGCP Endpoint name
mgcp-parm (ANY)	Matches the MGCP command parameter value.	MGCP Command Parameter
mgcp-rsp (ANY)	Matches the entire MGCP response line with the return code.	MGCP Reply Line
mgcp-rsp-000-line (ANY)	Matches the MGCP 0yz response acknowledgment.	MGCP 000 Reply Line
mgcp-rsp-100-line (ANY)	Matches the MGCP 1yz provisional response.	MGCP 100 Reply Line
mgcp-rsp-200-line (ANY)	Matches the MGCP 2yz successful completion response.	MGCP 200 Reply Line
mgcp-rsp-400-line (ANY)	Matches the MGCP 4yz permanent error response	MGCP 400 Reply Line
mgcp-rsp-500-line (ANY)	Matches the MGCP 5yz permanent error response.	MGCP 500 Reply Line
mgcp-rsp-800-line (ANY)	Matches the MGCP 8yz package-specific response codes.	MGCP 800 Reply Line
mgcp-rsp-bad-rcode (ANY)	Matches any MGCP invalid response code.	MGCP Invalid Response Code
mgcp-sdp-line (ANY)	Matches MGCP/SDP contents data line.	MGCP SDP Line
mgcp-trans-id (ANY)	Matches the MGCP transaction ID parameter value.	MGCP Transaction ID

Table 46: Service Contexts: Modbus

Context and Direction	Description	Display Name
modbus-except-resp (STC)	Matches a Modbus Exception Response.	Modbus Exception Response

Table 46: Service Contexts: Modbus (continued)

Context and Direction	Description	Display Name
modbus-requst (CTS)	Matches a Modbus Request	Modbus Request
modbus-response (STC)	Matches a Modbus Response.	Modbus Response
modbus-trailing-data (ANY)	Matches trailing data after the first MODBUS PDU.	Modbus Trailing Data

Table 47: Service Contexts: MSN

Context and Direction	Description	Display Name
msn-addrbook-url (STC)	Matches the URL for a user's address book.	MSN Addrbook Url
msn-compose-url (STC)	Matches the URL for composing an e-mail.	MSN Compose Url
msn-display-name (ANY)	Matches the display name of a user.	MSN Display Name
msn-get-file (STC)	Matches the name of a file that the client is downloading from a peer.	MSN Get File
msn-group-name (ANY)	Matches the name of a group of contacts.	MSN Group Name
msn-inbox-url (STC)	Matches the URL for a user's Inbox.	MSN Inbox Url
msn-ip-port (STC)	Matches the address and port of a switchboard server.	MSN IP Port
msn-message (ANY)	Matches the instant message text.	MSN Message
msn-message-application (ANY)	Matches the line of an application message (like file transfer).	MSN Message Application
msn-message-email-notification (STC)	Matches the line sent by the server to notify a client of new or unread e-mail.	MSN Message Email Notification
msn-message-header (ANY)	Matches the header line of an instant message.	MSN Message Header
msn-message-profile (STC)	Matches the line containing the profile of a message sender.	MSN Message Profile

Table 47: Service Contexts: MSN (continued)

Context and Direction	Description	Display Name
msn-passport-url (STC)	Matches login passport URL.	MSN Passport Url
msn-phone-number (ANY)	Matches the user's phone number.	MSN Phone Number
msn-png-chunk (ANY)	Matches contents of PNG chunk in MSN transaction.	MSN PNG CHUNK
msn-profile-url (STC)	Matches the URL of a user's passport profile.	MSN Profile Url
msn-put-file (CTS)	Matches the name of a file that the client is sending to a peer.	MSN Put File
msn-sign-in-name (ANY)	Matches the screen name (login name) of a user.	MSN Sign In Name
msn-url (STC)	Matches any URL in an MSN session	MSN URL
msn-user-state (ANY)	Matches the user's online state.	MSN User State

Table 48: Service Contexts: MSRPC

Context and Direction	Description	Display Name
msrpc-ans (STC)	Matches the response data in a MSRPC session	MSRPC ANSWER DATA
msrpc-call (CTS)	Matches the request data in a MSRPC session	MSRPC CALL DATA
msrpc-ifid-str (ANY)	Matches the interface ID string in an MSRPC session.	MSRPC IFID String
msrpc-raw (ANY)	Matches raw data in a MSRPC session	MSRPC RAW DATA

Table 49: Service Contexts: MS-SQL

Context and Direction	Description	Display Name
mssql-0x12 (CTS)	Matches the content of an MS-SQL type 0x12 request message.	MS-SQL 0x12 Request
mssql-cancel (CTS)	Matches the content of an MS-SQL cancel message	MS-SQL Cancel

Table 49: Service Contexts: MS-SQL (continued)

Context and Direction	Description	Display Name
mssql-login (CTS)	Matches thee content of an MS-SQL login message.	MS-SQL Login
mssql-login-app (CTS)	Matches the name of the application in an MS-SQL Login message.	MS-SQL Login Application
mssql-login-client (CTS)	Matches the name of the client in an MS-SQL Login message.	MS-SQL Login Client
mssql-login-database (CTS)	Matches the name of the database in an MS-SQL Login message.	MS-SQL Login Database
mssql-login-language (CTS)	Matches the name of the language in an MS-SQL Login message.	MS-SQL Login Language
mssql-login-lib (CTS)	Matches the name of the library in an MS-SQL Login message.	MS-SQL Login Library
mssql-login-pass (CTS)	Matches the password in an MS-SQL Login message.	MS-SQL Login Password
mssql-login-server (CTS)	Matches the name of the server in an MS-SQL Login message.	MS-SQL Login Server
mssql-login-user (CTS)	Matches the name of the user in an MS-SQL Login message.	MS-SQL Login User
mssql-query (CTS)	Matches the content of a MS-SQL query message.	MS-SQL Query
mssql-request-other (CTS)	Matches the content of an MS-SQL unknown Request message.	MS-SQL Request Other
mssql-rpe (CTS)	Matches the content of an MS-SQL RPC message.	MS-SQL RPC
mssql-rpc-name (CTS)	Matches the RPC name in an MS-SQL request message.	MS-SQL RPC Name

Table 50: Service Contexts: MySQL

Context and Direction	Description	Display Name
mysql-login-request-caps (CTS)	Matches the MYSQL Login Request Caps Data.	MS-SQL 0x12 Request
mysql-login-request-caps-pswd (CTS)	Matches the MYSQL Login Request Caps Password.	MS-SQL Cancel

Table 50: Service Contexts: MySQL (continued)

Context and Direction	Description	Display Name
mysql-login-request-caps-user (CTS)	Matches the MYSQL Login Request Caps Username.	MS-SQL Login
mysql-preamble (ANY)	Matches the 4 first bytes of the packet.	MS-SQL Login Application
mysql-requst-command (CTS)	Matches the MYSQL Request Command.	MS-SQL Login Client
mysql-response (STC)	Matches the MYSQL Response.	MS-SQL Login Database
mysql-server-greeting (STC)	Matches the MYSQL Server Greeting Data.	MS-SQL Login Language

Table 51: Service Contexts: NetBIOS

Context and Direction	Description	Display Name
nbds-browse-backup-server (ANY)	Matches the name of a backup server in a NetBIOS browse message.	NBDS Browse Backup Server
nbds-browse-server-name (ANY)	Matches the name of a server in a NetBIOS browse message.	NBDS Browse Server Name
nbds-destination-name (ANY)	Matches the destination name field in a NetBIOS message.	NBDS Destination Name
nbds-mailslot-name (ANY)	Matches the name of a mailslot in the NetBIOS mailslot message.	NBDS Mailslot Name
nbds-source-ip-address (ANY)	Matches the source IP field in the NetBIOS datagram header.	NBDS Source Ip Address
nbds-source-name (ANY)	Matches the source name field in a NetBIOS message.	NBDS Source Name
nbds-source-port (ANY)	Matches the source port fields in the NetBIOS datagram header.	NBDS Source Port
nbname-node-name (ANY)	Matches the node name in the status response message.	NBNAME Node Name
nbname-node-status (ANY)	Matches the statistics field of a node status response.	NBNAME Node Status

Table 51: Service Contexts: NetBIOS (continued)

Context and Direction	Description	Display Name
nbname-nsd-ip-address (ANY)	Matches the IP address of a NetBIOS name server specified in a redirect name query response message.	NBNAME Nsd IP Address
nbname-nsd-name (ANY)	Matches the name of a NetBIOS name server specified in a redirect name query response message.	NBNAME Nsd Name
nbname-resource-address (ANY)	Matches the IP address of a resource from the resource record.	NBNAME Resource Address
nbname-type-name (ANY)	Matches the type and name in a question or a resource record.	NBNAME Type Name

Table 52: Service Contexts: NFS

Context and Direction	Description	Display Name
nfs-create-name (CTS)	Matches the name of a file or directory in the CREATE procedure.	NFS Create Name
nfs-dir-entry (STC)	Matches the name of each directory entry returned by the READDIR procedure.	NFS Dir Entry
nfs-link-target (CTS)	Matches the name of the hard link in the LINK procedure.	NFS Link Target
nfs-lookup-name (CTS)	Matches the name of a file or directory in the LOOKUP procedure.	NFS Lookup Name
nfs-mkdir-name (CTS)	Matches the name of a directory in the MKDIR procedure.	NFS Mkdir Name
nfs-mknod-name (CTS)	Matches the name of the special file in the MKNOD procedure.	NFS Mknod Name
nfs-readlink-name (STC)	Matches the name returned by the READLINK procedure	NFS Readlink Name
nfs-remove-name (CTS)	Matches the name of a file in the REMOVE procedure.	NFS Remove Name
nfs-rename-from (CTS)	Matches the source file or directory name in the RENAME procedure.	NFS Rename From

Table 52: Service Contexts: NFS (continued)

Context and Direction	Description	Display Name
nfs-rename-to (CTS)	Matches the destination file or directory name in the RENAME procedure.	NFS Rename To
nfs-rmdir-name (CTS)	Matches the name of a directory in the RMDIR procedure.	NFS Rmdir Name
nfs-symlink-source (CTS)	Matches the source of the symbolic link in the SYMLINK procedure.	NFS Symlink Source
nfs-symlink-target (CTS)	Matches the target of the symbolic link in the SYMLINK procedure.	NFS Symlink Target

Table 53: Service Contexts: NNTP

Context and Direction	Description	Display Name
nntp-banner (STC)	Matches the NNTP banner.	NNTP Banner
nntp-body (ANY)	Matches each line of an NNTP message body.	NNTP Body
nntp-command-line (CTS)	Matches the entire NNTP command line.	NNTP Command Line
nntp-header (ANY)	Matches any header in an NNTP session.	NNTP Header
nntp-ihave-msgid (CTS)	Matches the message ID that appears in the IHAVE command of an NNTP session.	NNTP Ihave Msgid
nntp-mode (CTS)	Matches the NNTP mode.	NNTP Mode
nntp-msgid (ANY)	Matches the message ID that appears in various commands of an NNTP session.	NNTP Msgid
nntp-newsgroup (ANY)	Matches the name of news groups in an NNTP session.	NNTP Newsgroup

Table 54: Service Contexts: Normalized Stream

Context and Direction	Description	Display Name
normalized-stream (ANY)	Normalized Stream for services Telnet, IMAP, NFS, RPC, and Ruser only.	Normalized Stream

Table 54: Service Contexts: Normalized Stream (continued)

Context and Direction	Description	Display Name
normalized-stream1k (ANY)	Matches the first 1024 bytes of reassembled, normalized TCP stream data.	Normalized Stream 1K
normalized-stream256 (ANY)	Matches the first 256 bytes of reassembled, normalized TCP stream data.	Normalized Stream 256
normalized-stream8k (ANY)	Matches the first 8192 bytes of reassembled, normalized TCP stream data.	Normalized Stream 8K

Table 55: Service Contexts: NTP

Context and Direction	Description	Display Name
ntp-ctrl-data-opt (ANY)	Matches the data field in an NTP control message.	NTP Ctrl Data Opt
ntp-ctrl-opcode-response-var (ANY)	Matches each of the name and value pairs found in the NTP control message data field. The context includes a 1-byte NTP control message opcode and a 1-byte NTP response type.	NTP Ctrl Opcode Response Var

Table 56: Service Contexts: Packet

Context and Direction	Description	Display Name
packet (ANY)	Matches any packet in a session.	Packet

Table 57: Service Contexts: POP3

Context and Direction	Description	Display Name
pop3-apop (CTS)	Matches the arguments of the APOP command in a POP3 session.	POP3 Apop
pop3-auth (CTS)	Matches the arguments of the AUTH command in a POP3 session.	POP3 Auth
pop3-command (CTS)	Matches each of the POP3 command names in a POP3 session.	POP3 Command
pop3-command-line (CTS)	Matches each command line in a POP3 session.	POP3 Command Line

Table 57: Service Contexts: POP3 (continued)

Context and Direction	Description	Display Name
pop3-data-line (STC)	Matches lines in the e-mail body of an POP3 transaction.	POP3 Data Line
pop3-data-text-html (STC)	Matches lines in a text/html MIME attachment in the body of an POP3 transaction.	POP3 Data Text Html
pop3-data-text-plain (STC)	Matches lines in a text/plain MIME attachment in the body of an POP3 transaction.	POP3 Data Text Plain
pop3-dele (CTS)	Matches the arguments of the DELE command in a POP3 session.	POP3 Dele
pop3-header-comment (STC)	Matches the Comment: header of an e-mail in a POP3 transaction.	POP3 Header Comment
pop3-header-from (STC)	Matches the From: header of an e-mail in a POP3 transaction.	POP3 Header Comment
pop3-header-line (STC)	Matches each header line of an e-mail in POP3 transaction.	POP3 Header Line
pop3-header-reply-to (STC)	Matches the Reply-To: header of an e-mail in a POP3 transaction.	POP3 Header Reply To
pop3-header-sender (STC)	Matches the Sender: header of an e-mail in a POP3 transaction.	POP3 Header Sender
pop3-header-subject (STC)	Matches the Subject: header of an e-mail in a POP3 transaction	POP3 Header Subject
pop3-header-to (STC)	Matches the To: header of an e-mail in a POP3 transaction.	POP3 Header To
pop3-header-x-field (STC)	Matches each extended header (that start with X-) of an e-mail in a POP3 transaction.	POP3 Header X Field
pop3-header-x-mailer (STC)	Matches the X-Mailer: header of an e-mail in a POP3 transaction.	POP3 Header X Mailer
pop3-list (CTS)	Matches the arguments of the LIST command in a POP3 session.	POP3 List
pop3-mime-content-data (STC)	Matches the first 64 bytes of the base-64 decoded MIME attachment data in a POP3 session.	POP3 Mime Content Data

Table 57: Service Contexts: POP3 (continued)

Context and Direction	Description	Display Name
pop3-mime-content-filename (STC)	Matches the content filename of a MIME attachment in a POP3 session.	POP3 Mime Content Filename
pop3-mime-content-name (STC)	Matches the content name of a MIME attachment in a POP3 session.	POP3 Mime Content Name
pop3-retr (CTS)	Matches the arguments of the RETR command in a POP3 session.	POP3 Retr
pop3-top (CTS)	Matches the arguments of the TOP command in a POP3 session.	POP3 Top
pop3-uidl (CTS)	Matches the arguments of the UIDL command in a POP3 session.	POP3 Uidl
pop3-user (CTS)	Matches the user name of a POP3 session.	POP3 User
pop3-xtnd (CTS)	Matches the arguments of the XTND command in a POP3 session.	POP3 Xtnd

Table 58: Service Contexts: RADIUS

Context and Direction	Description	Display Name
radius-access-accept (STC)	Matches the attribute fields of a RADIUS Access-Accept message.	RADIUS Access Accept
radius-access-challenge (STC)	Matches the attribute fields of a RADIUS Access-Challenge message.	RADIUS Access Challenge
radius-access-reject (STC)	Matches the attribute fields of a RADIUS Access-Reject message.	RADIUS Access Reject
radius-access-request (CTS)	Matches the attribute fields of a RADIUS Access-Request message.	RADIUS Access Request
radius-acct-request (CTS)	Matches the attribute fields of a RADIUS Accounting-Request message.	RADIUS Acct Request
radius-acct-response (STC)	Matches the attribute fields of a RADIUS Accounting-Response message.	RADIUS Acct Response

Table 58: Service Contexts: RADIUS (continued)

Context and Direction	Description	Display Name
radius-attr-acct-multi-session-id (CTS)	Matches the value of an Account-Multi-Session-Id attribute.	RADIUS Attr Acct Multi Session Id
radius-attr-acct-session-id (CTS)	Matches the value of an Account-Session-Id attribute.	RADIUS Attr Acct Session Id
radius-attr-acct-tunnel-connection (CTS)	Matches the value of an Account-Tunnel-Connection attribute.	RADIUS Attr Acct Tunnel Connection
radius-attr-arap-features (STC)	Matches the value of an ARAP-Features attribute.	RADIUS Attr Arap Features
radius-attr-arap-password (CTS)	Matches the value of an ARAP-Password attribute.	RADIUS Attr Arap Password
radius-attr-arap-security-data (ANY)	Matches the value of an ARAP-Security-Data attribute.	RADIUS Attr Arap Security Data
radius-attr-callback-number (ANY)	Matches the value of a Callback-Number attribute.	RADIUS Attr Callback Number
radius-attr-called-station-id (CTS)	Matches the value of a Caller-Station-Id attribute.	RADIUS Attr Called Station Id
radius-attr-calling-station-id (CTS)	Matches the value of a Calling-Station-Id attribute.	RADIUS Attr Calling Station Id
radius-attr-chap-challenge (CTS)	Matches the value of a Chap-Challenge attribute.	RADIUS Attr Chap Challenge
radius-attr-chap-password (CTS)	Matches the value of a Chap-Password attribute.	RADIUS Attr Chap Password
radius-attr-configuration-token (STC)	Matches the value of a Configuration-Token attribute.	RADIUS Attr Configuration Token
radius-attr-connect-info (CTS)	Matches the value of a Connect-Info attribute.	RADIUS Attr Connect Info
radius-attr-eap-message (ANY)	Matches the value of an EAP-Message attribute.	RADIUS Attr Eap Message
radius-attr-filter-id (ANY)	Matches the value of a Filter-Id attribute.	RADIUS Attr Filter Id
radius-attr-framed-appletalk-zone (ANY)	Matches the value of a Framed-Appletalk-Zone attribute.	RADIUS Attr Framed Appletalk Zone

Table 58: Service Contexts: RADIUS (continued)

Context and Direction	Description	Display Name
radius-attr-framed-pool (STC)	Matches the value of a Framed-Pool attribute.	RADIUS Attr Framed Pool
radius-attr-framed-route (ANY)	Matches the value of a Framed-Route attribute.	RADIUS Attr Framed Route
radius-attr-login-lat-group (ANY)	Matches the value of a Login-LAT-Group attribute.	RADIUS Attr Login Lat Group
radius-attr-login-lat-node (ANY)	Matches the value of a Login-LAT-Node attribute.	RADIUS Attr Login Lat Node
radius-attr-login-lat-port (ANY)	Matches the value of a Login-LAT-Port attribute.	RADIUS Attr Login Lat Port
radius-attr-login-lat-service (ANY)	Matches the value of a Login-LAT-Service attribute.	RADIUS Attr Login Lat Service
radius-attr-message-authenticator (ANY)	Matches the value of a Message-Authenticator attribute.	RADIUS Attr Message Authenticator
radius-attr-nas-identifier (CTS)	Matches the value of a NAS-Identifier attribute.	RADIUS Attr Nas Identifier
radius-attr-nas-port-id (CTS)	Matches the value of a NAS-Port-Id attribute.	RADIUS Attr Nas Port Id
radius-attr-proxy-state (ANY)	Matches the value of a Proxy-State attribute.	RADIUS Attr Proxy State
radius-attr-reply-message (STC)	Matches the value of a Reply-Message attribute.	RADIUS Attr Reply Message
radius-attr-state (ANY)	Matches the value of a State attribute	RADIUS Attr State
radius-attr-tunnel-assignment-id (ANY)	Matches the value of a Tunnel-Assignemnt-Id attribute.	RADIUS Attr Tunnel Assignment Id
radius-attr-tunnel-client-auth-id (ANY)	Matche the value of a Tunnel-Client-Auth-Id attribute	RADIUS Attr Tunnel Client Auth Id
radius-attr-tunnel-client-endpoint (ANY)	Matches the value of a Tunnel-Client-Endpoint attribute.	RADIUS Attr Tunnel Client Endpoint
radius-attr-tunnel-password (STC)	Matches the value of a Tunnel-Password attribute.	RADIUS Attr Tunnel Password
radius-attr-tunnel-private-group-id (ANY)	Matches the value of a Tunnel-Private-Group-Id attribute.	RADIUS Attr Tunnel Private Group Id

Table 58: Service Contexts: RADIUS (continued)

Context and Direction	Description	Display Name
radius-attr-tunnel-server-auth-id (ANY)	Matche the value of a Tunnel-Server-Auth-Id attribute.	RADIUS Attr Tunnel Server Auth Id
radius-attr-tunnel-server-endpoint (ANY)	Matches the value of a Tunnel-Server-Endpoint attribute.	RADIUS Attr Tunnel Server Endpoint
radius-attr-user-name (ANY)	Matches the value of a User-Name attribute.	RADIUS Attr User Name
radius-attr-user-password (CTS)	Matches the value of a User-Password attribute.	RADIUS Attr User Password
radius-attr-vendor-specific (ANY)	Matches the value of a Vendor-Specific attribute.	RADIUS Attr Vendor Specific
radius-attribute (ANY)	Matches any RADIUS attribute, including the type, length and value.	RADIUS Attribute

Table 59: Service Contexts: REXEC

Context and Direction	Description	Display Name
rexec-remote-command (CTS)	Matches the remote command in an REXEC session.	REXEC Remote Command
rexec-remote-user (CTS)	Matches the remote username in an REXEC session.	REXEC Remote Username

Table 60: Service Contexts: RLOGIN

Context and Direction	Description	Display Name
rlogin-local-user (CTS)	Matches the local username in an RLOGIN session.	RLOGIN Local Username
rlogin-remote-user (CTS)	Matches the remote username in an RLOGIN session.	RLOGIN Remote Username

Table 61: Service Contexts: RSH

Context and Direction	Description	Display Name
rsh-local-user (CTS)	Matches the local username in an RSH session.	RSH Local Username
rsh-remote-command (CTS)	Matches the remote command in an RSH session.	RSH Remote Command

Table 61: Service Contexts: RSH (continued)

Context and Direction	Description	Display Name
rsh-remote-user (CTS)	Matches the remote username in an RSH session.	RSH Remote Username

Table 62: Service Contexts: RUSERS

Context and Direction	Description	Display Name
rusers-device (STC)	Matches the name of the device in an RUSERS session.	RUSERS Device
rusers-host (STC)	Matches the name of the host in an RUSERS session.	RUSERS Host
rusers-user (STC)	Matches the name of the user in an RUSERS session.	RUSERS User

Table 63: Service Contexts: SIP

Context and Direction	Description	Display Name
sip-bad-header (ANY)	Matches SIP hearders with bad syntax.	SIP BAD HEADER
sip-command-state (ANY)	Matches the state of the SIP connection.	SIP Command State
sip-content-any (ANY)	Matches SIP contents portion of packet data.	SIP CONTENT ANY
sip-content-sdp (ANY)	Matches SIP/SDP content data.	SIP CONTENT SDP
sip-display-name (ANY)	Matches the display name of URL in related headers.	SIP DISPLAY NAME
sip-header-any (ANY)	Matches SIP headers with no designated context.	SIP HEADER ANY
sip-header-callid (ANY)	Matches the SIP <Call-ID> header.	SIP HEADER CALLID
sip-header-from (ANY)	Matches the SIP <From> header.	SIP HEADER FROM
sip-header-maxforwards (CTS)	Matches the SIP <Max-Forwards> header.	SIP HEADER MAXFORWARDS
sip-header-to (ANY)	Matches SIP <To> header.	SIP HEADER TO

Table 63: Service Contexts: SIP (continued)

Context and Direction	Description	Display Name
sip-header-value-len (ANY)	Artificially created context for putting thresholds on a header value.	SIP HEADER VALUE LENGTH
sip-headr-via (ANY)	Matches the SIP <Via> header.	SIP HEADER VIA
sip-parameter (ANY)	Matches parsed parameters in the headers.	SIP PARAMETER
sip-parameter-bad (ANY)	Matches parsed invalid parameters in the headers.	SIP PARAMETER BAD
sip-reply (STC)	Matches any SIP reply line with the return code.	SIP Reply Line
sip-reply-100-line (STC)	Matches the SIP 1yz Positive Preliminary reply.	SIP Reply 100 Line
sip-reply-200-line (STC)	Matches the SIP 2yz Positive Compleation reply.	SIP Reply 200 Line
sip-reply-300-line (STC)	Matches the SIP 3yz Postive Intermediate reply.	SIP Reply 300 Line
sip-reply-400-line (STC)	Matches the SIP 4yz Transient Negative Completion reply.	SIP Reply 400 Line
sip-reply-500-line (STC)	Matches the SIP 5yz Permanent Negative Completion reply.	SIP Reply 500 Line
sip-reply-600-line (STC)	Matches the SIP 6yz Failure Completion reply.	SIP Reply 600 Line
sip-reply-bad-rcode (STC)	Matches any SIP invalid response code.	SIP Reply BAD RCODE
sip-request (CTS)	Matches the SIP request command line.	SIP Request
sip-request-unknown (CTS)	Matches the SIP request with unknown command.	SIP Unknown Request
sip-sdp-line (ANY)	Matches the SIP/SDP contents data line.	SIP SDP LINE
sip-unknown-data (ANY)	Matches SIP unknown data.	SIP UNKNOWN DATA
sip-unknown-header (ANY)	Matches a SIP unknown header.	SIP Unknown HEADER

Table 63: Service Contexts: SIP (continued)

Context and Direction	Description	Display Name
sip-uri-host (ANY)	Matches the host-name/IP-address of URI in related headers.	SIP URI HOST
sip-uri-parameter (ANY)	Matches the parameter of URI in related headers.	SIP URI PARAMETER

Table 64: Service Contexts: SMB

Context and Direction	Description	Display Name
smb-account-name (ANY)	Matches the SMB account name in the SESSION_SETUP_ANDX request of an SMB session.	SMB Account Name
smb-atsvc-request (CTS)	Matches any AT Service requests sent as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB AT Service Request
smb-atsvc-response (STC)	Matches any AT Service responses received as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB AT Service Response
smb-browser-request (CTS)	Matches any Browser requests sent as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	
smb-browser-response (STC)	Matches any Browser responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-called-name (ANY)	Matches the NetBIOS name of the initiator of an SMB session.	SMB Called Name
smb-calling-name (ANY)	Matches the NetBIOS name of the receiver of an SMB session.	SMB Calling Name
smb-connect-path (CTS)	Matches the connect path in the TREE_CONNECT_ANDX request of an SMB session.	SMB Connect Path

Table 64: Service Contexts: SMB (continued)

Context and Direction	Description	Display Name
smb-connect-service (CTS)	Matches the connect service in the TREE_CONNECT_ANDX request of an SMB session.	SMB Connect Service
smb-copy-filename (CTS)	Matches the filename in the COPY request of an SMB session.	SMB Copy Filename
smb-data (ANY)	Matches any SMB data portion.	SMB Data
smb-dce-rpc (ANY)	Matches any DCE/RPC message sent over the SMB Transport Layer.	SMB DCE/RPC
smb-dce-rpc-bind (CTS)	Matches any DCE/RPC bind message sent over the SMB Transport Layer.	SMB DCE/RPC Bind
smb-dce-rpc-bind-ack (STC)	Matches any DCE/RPC bind-ack message sent over the SMB Transport Layer.	SMB DCE/RPC Bind Ack
smb-dce-rpc-bind-nack (STC)	Matches any DCE/RPC bind-nack message sent over the SMB Transport Layer.	SMB DCE/RPC Bind Negative Ack
smb-dce-rpc-request (CTS)	Matches any DCE/RPC request message sent over the SMB Transport Layer.	SMB DCE/RPC Request
smb-dce-rpc-request-obj-uuid (CTS)	Matches object UUID of any DCE/RPC request message.	SMB DCE/RPC Request Object UUID
smb-dce-rpc-response (STC)	Matches any DCE/RPC response message sent over the SMB Transport Layer.	SMB DCE/RPC Response
smb-delete-filename (CTS)	Matches the filename in the DELETE request of an SMB session.	SMB Delete Filename
smb-dialect (CTS)	Matches each SMB dialect string in the NEGOTIATE request of an SMB session.	SMB Dialect
smb-lanman-request (CTS)	Matches any LANMAN requests sent as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB LANMAN Request

Table 64: Service Contexts: SMB (continued)

Context and Direction	Description	Display Name
smb-lanman-response (STC)	Matches any LANMAN responses received as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB LANMAN Response
smb-lsarpc-request (CTS)	Matches any Local Security Authority requests sent as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB Local Security Authority Request
smb-lsarpc-response (STC)	Matches any Local Security Authority responses received as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB Local Security Authority Request
smb-move-filename (CTS)	Matches the filename in the MOVE request of an SMB session.	SMB Move Filename
smb-native-lanman (ANY)	Matches the native LANMAN in the SESSION_SETUP_ANDX request of an SMB session.	SMB Native Lanman
smb-native-os (ANY)	Matches the native OS in the SESSION_SETUP_ANDX request of an SMB session.	SMB Native Os
smb-open-filename (CTS)	Matches the filename in the NT_CREATE_ANDX and OPEN_ANDX requests of an SMB session.	SMB Open Filename
smb-pipe-request (CTS)	Matches any generic named pipe requests over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB Named Pipe Request
smb-primary-domain (ANY)	Matches the SMB primary domain name in the SESSION_SETUP_ANDX request of an SMB session.	SMB Primary Domain

Table 64: Service Contexts: SMB *(continued)*

Context and Direction	Description	Display Name
smb-rename-filename (CTS)	Matches the filename in the RENAME request of an SMB session.	SMB Rename Filename
smb-samr-request (CTS)	Matches any Security Account Manager requests sent as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB Security Account Manager Request
smb-samr-response (STC)	Matches any Security Account Manager responses received as named pipe transactions over the SMB Transport Layer. The first 2 bytes of this context contains the opcode of the function.	SMB Security Account Manager Response
smb-spoolss-request (CTS)	Matches any Spool Subsystem requests sent as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	SMB Spool Subsystem Request
smb-spoolss-response (STC)	Matches any Spool Subsystem responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-srvsvc-request (CTS)	Matches any Server Service requests sent as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	SMB Server Service Request
smb-srvsvc-response (STC)	Matches any Server Service responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-svcctl-request (CTS)	Matches any Service Control Manager requests sent as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	SMB Service Control Manager Request

Table 64: Service Contexts: SMB (continued)

Context and Direction	Description	Display Name
smb-svcctl-response (STC)	Matches any Service Control Manager responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-trans2-create-directory (CTS)	Matches any SMB Transaction2 CREATE-DIRECTORY request.	
smb-trans2-request (CTS)	Matches any SMB Transaction2 request.	SMB Transaction2 Request
smb-trans2-response (STC)	Matches any SMB Transaction2 response.	SMB Transaction2 Response
smb-trans2-session-setup (CTS)	Matches any SMB Transaction2 SESSION-SETUP request.	
smb-trans2-set-file-info (CTS)	Matches any SMB Transaction2 SET-FILE-INFORMATION request.	
smb-trans2-set-path-info (CTS)	Matches any SMB Transaction2 SET-PATH-INFORMATION request.	SMB Transaction2 SET-PATH-INFO
smb-winreg-request (CTS)	Matches any Windows Remote Registry requests sent as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-winreg-response (STC)	Matches any Windows Remote Registry responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	
smb-wkssvc-request (CTS)	Matches any Workstation Service requests sent as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	

Table 64: Service Contexts: SMB (continued)

Context and Direction	Description	Display Name
smb-wkssvc-response (STC)	Matches any Workstation Service responses received as named pipe transactions over the SMB Transport Layer. The first two bytes of this context contains the opcode of the function.	

Table 65: Service Contexts: SMTP

Context and Direction	Description	Display Name
smtp-banner (STC)	Matches the banner returned by the server at the start of an SMTP transaction.	SMTP Banner
smtp-command-line (CTS)	Matches any SMTP command line.	SMTP Command Line
smtp-data-line (CTS)	Matches lines in the e-mail body of an SMTP transaction.	SMTP Data Line
smtp-data-text-html (CTS)	Matches lines in a text/html MIME attachment in the body of an SMTP transaction.	SMTP Data Text Html
smtp-data-text-plain (CTS)	Matches lines in a text/plain MIME attachment in the body of an SMTP transaction.	SMTP Data Text Plain
smtp-from (CTS)	Matches the contents of the MAIL, SAML, SEND, and SOML commands.	SMTP From
smtp-header (CTS)	Matches any unfolded header in the SMTP data.	SMTP Header
smtp-header-comment (CTS)	Matches the Comment: header in the SMTP data.	SMTP Header Comment
smtp-header-from (CTS)	Matches the From: header in the SMTP data.	SMTP Header From
smtp-header-line (CTS)	Matches any header lines in the SMTP data.	SMTP Header Line
smtp-header-reply-to (CTS)	Matches the Reply-To: header in the SMTP data.	SMTP Header Reply To
smtp-header-sender (CTS)	Matches the Sender: header in the SMTP data.	SMTP Header Sender

Table 65: Service Contexts: SMTP (continued)

Context and Direction	Description	Display Name
smtp-header-subject (CTS)	Matches the Subject: header in the SMTP data.	SMTP Header Subject
smtp-header-to (CTS)	Matches the To: header in the SMTP data.	SMTP Header To
smtp-header-x-field (CTS)	Matches all extended headers that start with X- in the SMTP data.	SMTP Header X Field
smtp-header-x-mailer (CTS)	Matches the X-Mailer: header in the SMTP data.	SMTP Header X Mailer
smtp-mime-content-data (CTS)	Matches the first 64 bytes of the base-64 decoded MIME attachment data in an SMTP session.	SMTP Mime Content Data
smtp-mime-content-filename (CTS)	Matches the content filename of a MIME attachment in an SMTP session.	SMTP Mime Content Filename
smtp-mime-content-name (CTS)	Matches the content name of a MIME attachment in an SMTP session.	SMTP Mime Content Name
smtp-pdf (ANY)	smtp-pdf	smtp-pdf
smtp-rcpt (CTS)	Matches the contents of the RCPT command in an SMTP transaction.	SMTP Rcpt
smtp-reply-100-line (STC)	Matches the SMTP 1yz Positive Preliminary reply.	SMTP Reply 100 Line
smtp-reply-200-line (STC)	Matches the SMTP 2yz Positive Completion reply.	SMTP Reply 200 Line
smtp-reply-300-line (STC)	Matches the SMTP 3yz Positive Intermediate reply.	SMTP Reply 300 Line
smtp-reply-400-line (STC)	Matches the SMTP 4yz Transient Negative Completion reply.	SMTP Reply 400 Line
smtp-reply-500-line (STC)	Matches the SMTP 5yz Permanent Negative Completion reply.	SMTP Reply 500 Line
smtp-reply-line (STC)	Matches the SMTP reply line.	SMTP Reply Line

Table 66: Service Contexts: SNMP

Context and Direction	Description	Display Name
snmp-community (ANY)	Matches the community name in any SNMP request or response.	SNMP Community
snmp-get-bulk-oid (CTS)	Matches the binary OID in any SNMP Get-Bulk request.	SNMP Get Bulk Oid
snmp-get-bulk-oid-parsed (CTS)	Matches the human-readable OID in any SNMP Get-Bulk request.	SNMP Get Bulk Oid Parsed
snmp-get-next-oid (CTS)	Matches the binary OID in any SNMP Get-Next request.	SNMP Get Next Oid
snmp-get-next-oid-parsed (CTS)	Matches the human-readable OID in any SNMP Get-Next request.	SNMP Get Next Oid Parsed
snmp-get-oid (CTS)	Matches the binary OID in any SNMP Get request.	SNMP Get Oid
snmp-get-oid-parsed (CTS)	Matches the human-readable OID in any SNMP Get request.	SNMP Get Oid Parsed
snmp-oid (ANY)	Matches the binary OID in any SNMP request or response.	SNMP Oid
snmp-oid-parsed (ANY)	Matches the human-readable OID in any SNMP request or response.	SNMP Oid Parsed
snmp-set-oid (CTS)	Matches the binary OID in any SNMP Set request.	SNMP Set Oid
snmp-set-oid-parsed (CTS)	Matches the human-readable OID in any SNMP Set request.	SNMP Set Oid Parsed
snmptrap-community (CTS)	Matches the community name in any SNMPTRAP message.	SNMPTRAP Community
snmptrap-eid (CTS)	Matches the binary EID (Enterprise-ID) in any SNMPTRAP message.	SNMPTRAP Eid
snmptrap-eid-parsed (CTS)	Matches the human-readable EID (Enterprise-ID) in any SNMPTRAP message.	SNMPTRAP Eid Parsed
snmptrap-inform-oid (CTS)	Matches the binary OID in any SNMPTRAP Inform message.	SNMPTRAP inform Oid

Table 66: Service Contexts: SNMP (continued)

Context and Direction	Description	Display Name
snmptrap-inform-oid-parsed (CTS)	Matches the human-readable OID in any SNMPTRAP Inform message.	SNMPTRAP Inform Oid Parsed
snmptrap-oid (CTS)	Matches the binary OID in any SNMPTRAP message.	SNMPTRAP Oid
snmptrap-oid-parsed (CTS)	Matches the human-readable OID in any SNMPTRAP message.	SNMPTRAP Oid Parsed
snmptrap-v2-oid (CTS)	Matches the binary OID in any SNMPTRAP v2 message.	SNMPTRAP v2 Oid
snmptrap-v2-oid-parsed (CTS)	Matches the human-readable OID in any SNMPTRAP v2 message.	SNMPTRAP v2 Oid Parsed

Table 67: Service Contexts: SSH

Display Name	Display Name	Display Name
ssh-header (ANY)	Matches the header at the start of an SSH session.	SSH Header

Table 68: Service Contexts: SSL

Context and Direction	Description	Display Name
ssl-cert-common-name (ANY)	Matches the common name attribute of the SSL certificate.	SSL Cert Common Name
ssl-cert-organization-name (ANY)	Matches the organization name in the SSL certificate.	SSL Cert Organization Name
ssl-cert-organizational-unit-name (ANY)	Matches the organizational unit name in the SSL certificate.	SSL Cert Organizational Unit Name
ssl-certificate (ANY)	Matches the entire SSL certificate content.	SSL Certificate
ssl-change-cipher-spec (ANY)	Matches the Change-Cipher-Spec Message Content	SSL Change Cipher Spec
ssl-client-hello (CTS)	Matches SSL client hello message content.	SSL_CLIENT_HELLO
ssl-client-key-exchange (CTS)	Matches SSL client key exchange message content.	SSL Client Key Exchange

Table 68: Service Contexts: SSL (continued)

Context and Direction	Description	Display Name
ssl-client-version (CTS)	Matches the client SSL version.	SSL Client Version
ssl-selected-cipher-suite (STC)	Matches the selected cipher suite in the server hello message.	SSL Selected Cipher Suite
ssl-server-hello (STC)	Matches SSL server hello message content.	SSL_SERVER_HELLO
ssl-server-key-exchange (STC)	Matches SSL server key exchange message content.	SSL Server Key Exchange
ssl-server-version (STC)	Matches the SSL server version.	SSL Server Version

Table 69: Service Contexts: Stream

Context and Direction	Description	Display Name
stream (ANY)	Matches the reassembled, normalized TCP stream data.	Stream
stream1k (ANY)	Matches the first 1024 bytes of reassembled TCP stream data.	Stream 1K
stream256 (ANY)	Matches the first 256 bytes of reassembled, normalized TCP stream data.	Stream 256
stream8k (ANY)	Matches the first 8192 bytes of reassembled TCP stream data.	Stream 8K

Table 70: Service Contexts: Telnet

Context and Direction	Description	Display Name
telnet-option (ANY)	Matches each of the telnet options in a Telnet session.	TELNET Option
telnet-subnegotiation (ANY)	Matches each of the telnet subnegotiation options in a Telnet session.	TELNET Subnegotiation
telnet-user (CTS)	Matches the Telnet user name.	TELNET User

Table 71: Service Contexts: TFTP

Context and Direction	Description	Display Name
tftp-filename (CTS)	Matches any filename in a TFTP session.	TFTP Filename
tftp-get-filename (CTS)	Matches the get filename in a TFTP session.	TFTP Get Filename
tftp-put-filename (CTS)	Matches the put filename in a TFTP session.	TFTP Put Filename

Table 72: Service Contexts: TNS

Context and Direction	Description	Display Name
tns-accept-section (STC)	Matches the Accept Section Data in a TNS session.	TNS Accept Section
tns-connect-addr-dev (CTS)	Matches the Connect Address-Dev in a TNS session.	TNS Connect Address-Dev
tns-connect-addr-host (CTS)	Matches the Connect Address-Host in a TNS session.	TNS Connect Address-Host
tns-connect-addr-key (CTS)	Matches the Connect Address-Key in a TNS session.	TNS Connect Address-Key
tns-connect-addr-port (CTS)	Matches the Connect Address-Port in a TNS session.	TNS Connect Address-Port
tns-connect-addr-proto (CTS)	Matches the Connect Address-Protocol in an TNS session.	TNS Connect Address-Protocol
tns-connect-cid-host (CTS)	Matches the Connect Data CID Host in a TNS session.	TNS Connect Data CID Host
tns-connect-cid-user (CTS)	Matches the Connect Data CID User in a TNS session.	TNS Connect Data CID User
tns-connect-data-cid-prog (CTS)	Matches the Connect Data CID Program in a TNS session.	TNS Connect Data CID Program
tns-connect-data-sid (CTS)	Matches the Connect Data SID in a TNS session.	TNS Connect Data SID
tns-connect-data-svcname (CTS)	Matches the Connect Data Service Name in an TNS session.	TNS Connect Data Service Name
tns-connect-section (CTS)	Matches the Connect Section Data in a TNS session.	TNS Connect Section

Table 72: Service Contexts: TNS (continued)

Context and Direction	Description	Display Name
tns-data-flags (ANY)	Matches 2 bytes flags of Data Section in an TNS session	TNS Data Flags
tns-data-section (ANY)	Matches the Data Section Data in a TNS session.	TNS Data Section
tns-message-body (ANY)	Matches any Message Body in a TNS session.	TNS Message Body
tns-message-type (ANY)	Matches the Message Type in a TNS session.	TNS Message Type
tns-preamble (ANY)	Matches the first 8 bytes of a TNS message.	TNS Preamble
tns-redirect-section (STC)	Matches the Redirect Section in a TNS session.	TNS Redirect Section

Table 73: Service Contexts: VNC

Context and Direction	Description	Display Name
vnc-client-version (CTS)	Matches the version number of the VNC protocol sent by the client.	VNC Client Version
vnc-reason (STC)	Matches the connection fail reason reported by the VNC server.	VNC Reason
vnc-server-version (STC)	Matches the version number of the VNC protocol sent by the server.	VNC Server Version

Table 74: Service Contexts: YMSG

Context and Direction	Description	Display Name
ymsg-alias (ANY)	Matches the alternate name associated with the main username.	YMSG Alias
ymsg-buddy-name (ANY)	Matches the name of a user that appears on the friends list.	YMSG Buddy Name
ymsg-chatroom-chatter (ANY)	Matches the name of a user participating in a chat session	YMSG Chatroom Chatter

Table 74: Service Contexts: YMSG (continued)

Context and Direction	Description	Display Name
ymsg-chatroom-invitee (ANY)	Matches the name of the user who is being invited to join a chatroom.	YMSG Chatroom Invitee
ymsg-chatroom-message (ANY)	Matches the messages exchanged in a chatroom.	YMSG Chatroom Name
ymsg-chatroom-name (ANY)	Matches the name of a chatroom in a YMSG session.	YMSG Chatroom Name
ymsg-conf-host (ANY)	Matches the name of the user who is hosting the conference.	YMSG Conf Host
ymsg-conf-invitee (ANY)	Matches the name of a user who is invited to a conference.	YMSG Config Url
ymsg-conf-join-msg (ANY)	Matches the content of a message sent as part of a conference invitation.	YMSG Conf Invitee
ymsg-conf-name (ANY)	Matches the name of a conference session.	YMSG Conf Join Msg
ymsg-config-url (STC)	Matches the URL at which the user can configure the password after the account is disabled.	YMSG Conf Name
ymsg-contact-name (ANY)	Matches the contact name in a friends list or invitation.	YMSG Contact Name
ymsg-group-name (ANY)	Matches the name of a group used to categorize friends.	YMSG Group Name
ymsg-header (ANY)	Matches data in the protocol header.	ymsg-header
ymsg-ignored-user (ANY)	Matches the name of the user being added to, or appearing on, the ignored users list.	YMSG Ignored User
ymsg-mail-sender (STC)	Matches the name of the user sending an e-mail message.	YMSG Mail Sender
ymsg-mail-sender-address (STC)	Matches the e-mail address of sender.	YMSG Mail Sender Address
ymsg-mail-subject (STC)	Matches the e-mail subject.	YMSG Mail Subject

Table 74: Service Contexts: YMSG (continued)

Context and Direction	Description	Display Name
ymsg-main-identity (ANY)	Matches the main identity name of the user.	YMSG Main Identity
ymsg-message (ANY)	Matches the instant message that is sent from one client to another.	YMSG Message
ymsg-message-server-filename-url (STC)	Matches the message with the name of the file on the client from which the server can download and transfer to peers.	YMSG Message Server Filename Url
ymsg-nickname (ANY)	Matches the nickname of a user.	YMSG Nickname
ymsg-p2p-get-filename (STC)	Matches the name of the file on the peer from which the file can be downloaded.	YMSG P2p Get Filename
ymsg-p2p-get-filename-url (STC)	Matches the location of a file on the peer from which the file can be downloaded.	YMSG P2p Get Filename Url
ymsg-p2p-put-filename (CTS)	Matches the name of the file on the client that other peers can download.	YMSG P2p Put Filename
ymsg-p2p-put-filename-url (CTS)	Matches the location of a file on the client from which other peers can download.	YMSG P2p Put Filename Url
ymsg-recipient (ANY)	Matches the identity of the recipient of a message or file.	YMSG Recipient
ymsg-sender (ANY)	Matches the identity of a sender of a message or file.	YMSG Sender
ymsg-server-get-filename-url (STC)	Matches the location of a file on the client from which the server can download and transfer to peers.	YMSG Server Get Filename Url
ymsg-system-message (STC)	Matches the content of a message sent from the server to the client.	YMSG System Message
ymsg-user-name (ANY)	Matches the identity of the login user or one of the user's alias.	YMSG User Name

Creating a Compound Attack Object

Use compound attack objects in cases where:

- Attacks use multiple methods to exploit a vulnerability and, inspected independently, the individual contexts appear benign.
- Matching multiple contexts reduces false positives.
- Coupling a signature with a protocol anomaly reduces false positives.

You select signature attack objects or predefined anomalies as “members” of the compound object, and you use Boolean expressions to specify matching logic.

To configure a compound attack object:

1. Configure general attack object properties and reference information as described for signature attack objects.

   On the Target Platform and Type page, select a target platform, select **Compound Attack**, and click **Next**.

2. On the Custom Attack – General Properties page, configure the settings described in Table 75 on page 161.

Table 75: Custom Attack – General Properties

Property	Description
Time Binding	Same guidelines as for signature attack objects.

   Click **Next**.

3. On the Compound Members page, specify compound attack parameters and add members. Table 76 on page 161 provides guidelines for completing the settings.

Table 76: Compound Attack Parameters

Setting	Description
Scope	Specify if the attack is matched within a session or across transactions in a session. Select one of the following: • **Session**–Allows multiple matches for the object within the same session. • **Transaction**–Matches the object across multiple transactions that occur within the same session.
Reset	Enable this option to generate a new log each time an attack is detected within the same session. If this option is not selected, then the attack is logged only once per session.

Table 76: Compound Attack Parameters (continued)

Setting	Description
Boolean Expression	Enter a Boolean expression of attack members used to identify the way attack members should be matched. Type a Boolean expression using the following Boolean operators: • OR–If either of the member name patterns match, the expression matches. • AND–If both of the member name patterns match, the expression matches. It does not matter which order the members appear in. • OAND–If both member name patterns match, and if they appear in the same order as in the Boolean expression, the expression matches. For example, the Boolean expression (s1 OAND s2) OR (s1 OAND s3)) AND (s4 AND s5) would match an attack that contains s1 followed by either s2 or s3, and that also contains s4 and s5 in any location.
Add member	Click the + icon, select **Signature** or **Protocol Anomaly**, and complete the configuration details. For signature members, specify the same contextual information as you do for a signature attack object. For protocol anomaly members, select from a list of predefined protocol anomalies. BEST PRACTICE: Example of the naming convention for members are: m01, m02, m03, and so on. It is recommend to use this same naming convention.
Order	Enable this option to create a compound attack object that must match each member signature or protocol anomaly in the order you specify. If you do not specify an order, the compound attack object still must match all members, but the pattern or protocol anomalies can appear in the attack in any order. A compound attack object detects attacks that use multiple methods to exploit a vulnerability.
Protocol Binding	Protocol binding over which attack will be detected.

4. Click **Finish**.

See Also

- Creating a Signature Attack Object on page 81
- Testing a Custom Attack Object on page 80

Modifying Custom Attack Objects Due to Changes Introduced in Signature Update

This topic describes changes to some service contexts generated by the HTTP protocol decoder. Beginning with Signature Update #1972, the HTTP protocol decoder no longer generates some contexts. If your IDP security policy includes custom signatures that use the contexts that have been removed, you must modify your attack object definitions as described below to avoid policy compilation errors. This topic includes the following information:

Reference: Removed Contexts

To improve performance, the HTTP protocol decoder no longer generates the contexts listed in the first column of Table 77 on page 163. Review this table for guidelines on replacing the contexts in custom attack objects.

Table 77: HTTP Service Contexts

Removed	Replace With	Guideline
http-text-html-body	http-text-html	Change signatures that use context http-text-html-body to http-text-html. You do not need to make changes to the signature pattern or other properties.
• http-get-url-parsed-param • http-post-url-parsed-param • http-head-url-parsed-param • http-get-url-parsed-param-parsed • http-post-url-parsed-param-parsed • http-head-url-parsed-param-parsed	Use a combination of the following contexts: • http-request-method • http-url-parsed • http-variable-parsed	Use a compound signature with a Boolean AND to break the signature pattern into multiple pieces. Ensure the Scope field is set to Transaction. Using the http-request-method context is optional. You use the http-request-method context to bind detection to http GET or POST or HEAD transactions. For GET method, we use the pattern \[GET\] (case insensitive GET). Use http-request-method only if the results you logged previously matching on Request Method are worth preserving. If not, omit it to improve performance. If you use http-request-method, order it first in the compound chain. Use the http-url-parsed context to match an attack signature identifiable in the URL. Use this context to match a pattern in the URL that appears before variable parameters—the part of the URL before the question mark (?). Use one or more http-variable-parsed contexts to match the URL variable parameters—the part of the URL after the question mark (?), normally separated by ampersands (&).

Example: Replacing the Context for Patterns Appearing in HTML Text

Each context generated by the HTTP detector engine has a performance cost. Contexts http-text-html and http-text-html-body serve the same purpose. Reducing the number of contexts improves performance.

Table 78 on page 163 shows the properties of a signature before Update #1972 and the signature after. This is a simple change. You change only the context. You do not need to change the pattern or other properties.

Table 78: HTTP Service Contexts: HTML Text

	Before Update	After Update
Context	http-text-html-body	http-text-html
Pattern	**.*<span></span>.***	**.*<span></span>.***

Example: Replacing the Contexts for Patterns Appearing in URLs

This section has two parts:

- Signatures that Match Request Methods on page 164
- Signatures that Match URL Strings and URL Variables on page 164

Signatures that Match Request Methods

When modifying custom attack objects that previously matched request methods GET, POST, or HEAD, consider whether matches against these request method patterns were effective for you. Keep in mind, each context generated has a performance cost. If request method is not essential to your results, take this opportunity to recast your signature without it.

Table 79 on page 164 and Table 80 on page 164 show the properties of a signature before Update #1972 and the compound signature after. This example preserves an interest in request method.

Table 79: HTTP Service Contexts: Request Methods Before Update

	Signature Before Update
Scope	–
Context	http-get-url-parsed-param
Pattern	**\[/viper/vegaspalms/\].***

Table 80: HTTP Service Contexts: Request Methods After Update

	Compound Signature After Update	
	m01	m02
Scope	Transaction	
Context	http-request-method	http-url-parsed
Pattern	**\[GET\]**	**\[/viper/vegaspalms/\].***

Signatures that Match URL Strings and URL Variables

In general, breaking a single pattern into multiple contexts could positively or negatively impact performance. You need to test your changes to understand performance impact before deploying the attack objects in a production network. The example shown in Table 81 on page 165 and Table 82 on page 165 breaks URL matching into multiple contexts. Our security team has tested performance for the recommendations described here.

Table 81: HTTP Service Contexts: URL Strings and Variables Before Update

	Signature Before Update
Scope	–
Context	http-get-url-param-parsed-param
Pattern	\[/cvs/index[0-9]?\.php\?option=com_content&do_pdf=1&id=1\]

Table 82: HTTP Service Contexts: URL Strings and Variables After Update

	Compound Signature After Update			
	m01	m02	m03	m04
Scope	Transaction			
Context	http-url-parsed	http-variable-parsed	http-variable-parsed	http-variable-parsed
Pattern	\[/cvs/index[0-9]?\.php\]	\[option=com_content\]	\[do_pdf=1\]	\[id=1\]

See Also

- Creating a Compound Attack Object on page 161
- Testing a Custom Attack Object on page 80

Example: Configuring Compound or Chain Attacks

This example shows how to configure compound or chain attacks for specific match criteria. A compound or chain attack object can be configured to detect attacks that use multiple methods to exploit a vulnerability.

- Requirements on page 165
- Overview on page 165
- Configuration on page 166
- Verification on page 170

Requirements

Before you begin, IDP must be supported and enabled on the device.

Overview

A compound or a chain attack object can combine the signatures and anomalies to form a single attack object. A single attack object can contain:

- Two or more signatures
- Two or more anomalies
- A combination of signatures and anomalies

Compound or chain attack objects combine multiple signatures and/or protocol anomalies into a single attack object, forcing traffic to match a pattern of combined signatures and anomalies within the compound attack object before traffic is identified as an attack. These objects are also used to reduce false positives and to increase detection accuracy. It enables you to be specific about the events that need to occur before IDP identifies traffic as an attack.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy idpengine rulebase-ips rule 1 match from-zone any
set security idp idp-policy idpengine rulebase-ips rule 1 match source-address any
set security idp idp-policy idpengine rulebase-ips rule 1 match to-zone any
set security idp idp-policy idpengine rulebase-ips rule 1 match destination-address any
set security idp idp-policy idpengine rulebase-ips rule 1 match application default
set security idp idp-policy idpengine rulebase-ips rule 1 match attacks custom-attacks
  ftpchain
set security idp idp-policy idpengine rulebase-ips rule 1 then action no-action
set security idp idp-policy idpengine rulebase-ips rule 1 then notification log-attacks
set security idp active-policy idpengine
set security idp custom-attack ftpchain severity info
set security idp custom-attack ftpchain attack-type chain protocol-binding application
  ftp
set security idp custom-attack ftpchain attack-type chain scope session
set security idp custom-attack ftpchain attack-type chain order
set security idp custom-attack ftpchain attack-type chain member m1 attack-type
  signature context ftp-banner
set security idp custom-attack ftpchain attack-type chain member m1 attack-type
  signature pattern .*vsFTPd.*
set security idp custom-attack ftpchain attack-type chain member m1 attack-type
  signature direction server-to-client
set security idp custom-attack ftpchain attack-type chain member m2 attack-type
  signature context ftp-username
set security idp custom-attack ftpchain attack-type chain member m2 attack-type
  signature pattern .*root.*
set security idp custom-attack ftpchain attack-type chain member m2 attack-type
  signature direction client-to-server
set security idp custom-attack ftpchain attack-type chain member m3 attack-type
  anomaly test LOGIN_FAILED
set security idp custom-attack ftpchain attack-type chain member m3 attack-type
  anomaly direction any
set security idp traceoptions file idpd
set security idp traceoptions flag all
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure compound or chain attacks for specific match criteria:

1. Create an IDP policy.

   ```
   [edit]
   user@host# set security idp idp-policy idpengine
   ```

2. Associate a rulebase with the policy.

   ```
   [edit security idp idp-policy idpengine]
   user@host# edit rulebase-ips
   ```

3. Add rules to the rulebase.

   ```
   [edit security idp idp-policy idpengine rulebase-ips]
   user@host# edit rule 1
   ```

4. Define the match criteria for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match from-zone any
   user@host# set match source-address any
   user@host# set match to-zone any
   user@host# set match destination-address any
   ```

5. Specify an application set name to match the rule criteria.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match application default
   ```

6. Specify the match attack object and name for the attack object.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match attacks custom-attacks ftpchain
   ```

7. Specify an action for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set then action no-action
   ```

8. Specify notification or logging options for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set then notification log-attacks
   ```

9. Activate the IDP policy.

   ```
   [edit]
   user@host# set security idp active-policy idpengine
   ```

10. Specify a name for the custom attack.

    ```
    [edit security idp]
    user@host# set custom-attack ftpchain
    ```

11. Set the severity for the custom attack.

    ```
    [edit security idp custom-attack ftpchain]
    user@host# set severity info
    ```

12. Set the attack type and the application name for the custom attack.

    ```
    [edit security idp custom-attack ftpchain]
    user@host# set attack-type chain protocol-binding application ftp
    ```

13. Set the scope and the order in which the attack is defined.

    ```
    [edit security idp custom-attack ftpchain attack-type chain]
    user@host# set scope session
    user@host# set order
    ```

14. Specify a name for the first member of the chain attack object.

    ```
    [edit security idp custom-attack ftpchain attack-type chain]
    user@host# set member m1
    ```

15. Set the context, pattern, and direction for the first member of the chain attack object.

    ```
    [edit security idp custom-attack ftpchain attack-type chain member m1]
    user@host# set attack-type signature context ftp-banner
    user@host# set attack-type signature pattern .*vsFTPd.*
    user@host# set attack-type signature direction server-to-client
    ```

16. Specify a name for the second member of the chain attack object.

    ```
    [edit security idp custom-attack ftpchain attack-type chain]
    user@host# set member m2
    ```

17. Set the context, pattern, and direction for the second member of the chain attack object.

    ```
    [edit security idp custom-attack ftpchain attack-type chain member m2]
    user@host# set attack-type signature context ftp-username
    user@host# set attack-type signature pattern .*root.*
    user@host# set attack-type signature direction client-to-server
    ```

18. Specify a name for the third member of the chain attack object.

    ```
    [edit security idp custom-attack ftpchain attack-type chain]
    user@host# set member m3
    ```

19. Specify an attack-type and direction for the third member of the chain attack object.

```
[edit security idp custom-attack ftpchain attack-type chain member m3]
user@host# set attack-type anomaly direction any
```

20. Specify the trace options and trace file information for the IDP services.

```
[edit]
user@host# set security idp traceoptions file idpd
```

21. Specify the events and other information which needs to be included in the trace output.

```
[edit]
user@host# set security idp traceoptions flag all
```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy idpengine {
  rulebase-ips {
    rule 1 {
      match {
        from-zone any;
        source-address any;
        to-zone any;
        destination-address any;
        application default;
        attacks {
          custom-attacks ftpchain;
        }
      }
      then {
        action {
          no-action;
        }
        notification {
          log-attacks;
        }
      }
    }
  }
}
active-policy idpengine;
custom-attack ftpchain {
  severity info;
  attack-type {
    chain {
      protocol-binding {
        application ftp;
      }
      scope session;
```

```
        order;
        member m1 {
          attack-type {
            signature {
              context ftp-banner;
              pattern .*vsFTPd.*;
              direction server-to-client;
            }
          }
        }
        member m2 {
          attack-type {
            signature {
              context ftp-username;
              pattern .*root.*;
              direction client-to-server;
            }
          }
        }
        member m3 {
          attack-type {
            anomaly {
              test LOGIN_FAILED;
              direction any;
            }
          }
        }
      }
    }
  }
  traceoptions {
    file idpd;
    flag all;
  }
```

If you are done configuring the device, enter **commit** from configuration mode.

NOTE: When you enter **commit** in configuration mode, the configuration is internally verified and then committed. If there are any errors, commit will fail and the errors will be reported.

Verification

To confirm that the chain attack configuration is working properly, perform this task:

- Verifying the Configuration on page 170

Verifying the Configuration

Purpose Verify that the chain attack configuration is correct.

Action From operational mode, enter the **show security idp policy-commit-status** command to check the policy compilation or load status.

NOTE: The output of the **show security idp policy-commit-status** command is dynamic, hence there is no single output for this command.

Verify that the attacks are getting detected as per the configuration, pass traffic through the device to trigger an attack match. For example, enter the **show security idp status** command to check whether the policy is loaded or not.

```
user@host> show security idp status

  IDP policy[/var/db/idpd/bins/test.bin.gz.v] and
    detector[/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v]
    loaded successfully.
  The loaded policy size is:785 Bytes
```

Enter the **show security idp attack table** command to pass attack traffic and then verify that the attacks are getting detected or not.

NOTE: The command will display the output only when attacks are detected.

```
user@host> show security idp attack table

  IDP attack statistics:
  Attack name #Hits
  FTP:USER:ROOT 1
```

Example: Configuring Attack Groups with Dynamic Attack Groups and Custom Attack Groups

This example shows how to configure attack groups with dynamic attack groups and custom attack groups in an IDP policy to protect an FTP or Telnet server.

- Requirements on page 171
- Overview on page 172
- Configuration on page 172
- Verification on page 177

Requirements

Before you begin, install the security package on the device only if one of the following statements is true:

- Dynamic attack groups are configured.
- Custom attack groups contain predefined attacks or attack groups.

NOTE: If custom attack groups contain only custom attacks, the security package license is not required and the security package need not be installed on the device. To install the security package, you need an IDP security package license.

See "Understanding IDP Policy Rules" on page 55.

Overview

IDP contains a large number of predefined attack objects. To manage and organize IDP policies, attack objects can be grouped. An attack object group can contain two or more types of attack objects. The attack groups are classified as follows:

- Dynamic attack group—Contains attack objects based on certain matching criteria. During a signature update, dynamic group membership is automatically updated based on the matching criteria for that group. For example, you can dynamically group the attacks related to a specific application using the dynamic attack group filters.
- Custom attack group—Contains a list of attacks that are specified in the attack definition. A custom attack group can also contain specific predefined attacks, custom attacks, predefined attack groups, or dynamic attack groups. A custom attack group is static in nature as the attacks are specified in the group. Therefore, the attack group do not change when the security database is updated. The members can be predefined attacks or predefined attack groups from the signature database or other custom attacks and dynamic attack groups.

In this example we configure an attack group in an IDP policy to protect an FTP or Telnet server against custom and dynamic attacks.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy idpengine rulebase-ips rule 1 match from-zone any
set security idp idp-policy idpengine rulebase-ips rule 1 match source-address any
set security idp idp-policy idpengine rulebase-ips rule 1 match to-zone any
set security idp idp-policy idpengine rulebase-ips rule 1 match destination-address any
set security idp idp-policy idpengine rulebase-ips rule 1 match application default
set security idp idp-policy idpengine rulebase-ips rule 1 match attacks
  custom-attack-groups cust-group
set security idp idp-policy idpengine rulebase-ips rule 1 match attacks
  dynamic-attack-groups dyn2
set security idp idp-policy idpengine rulebase-ips rule 1 then action no-action
set security idp idp-policy idpengine rulebase-ips rule 1 then notification log-attacks
set security idp active-policy idpengine
set security idp custom-attack customftp severity info
set security idp custom-attack customftp attack-type signature context ftp-username
set security idp custom-attack customftp attack-type signature pattern .*guest.*
set security idp custom-attack customftp attack-type signature direction client-to-server
set security idp custom-attack-group cust-group group-members customftp
set security idp custom-attack-group cust-group group-members ICMP:INFO:TIMESTAMP
set security idp custom-attack-group cust-group group-members "TELNET - Major"
set security idp custom-attack-group cust-group group-members dyn1
set security idp dynamic-attack-group dyn1 filters category values TROJAN
set security idp dynamic-attack-group dyn2 filters direction expression and
set security idp dynamic-attack-group dyn2 filters direction values server-to-client
set security idp dynamic-attack-group dyn2 filters direction values client-to-server
set security idp traceoptions file idpd
set security idp traceoptions flag all
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure attack groups with dynamic attack groups and custom attack groups:

1. Create an IDP policy.

   ```
   [edit]
   user@host# set security idp idp-policy idpengine
   ```

2. Associate a rulebase with the policy.

   ```
   [edit security idp idp-policy idpengine]
   user@host# set rulebase-ips
   ```

3. Add rules to the rulebase.

   ```
   [edit security idp idp-policy idpengine rulebase-ips]
   user@host# set rule 1
   ```

4. Define the match criteria for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match from-zone any
   user@host# set match source-address any
   user@host# set match to-zone any
   user@host# set match destination-address any
   ```

5. Specify an application set name to match the rule criteria.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match application default
   ```

6. Specify a match for the custom attack group.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set match attacks custom-attack-groups cust-group
   ```

7. Specify a match for the dynamic attack group.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   ```

```
user@host# set match attacks dynamic-attack-groups dyn2
```

8. Specify an action for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set then action no-action
   ```

9. Specify notification or logging options for the rule.

   ```
   [edit security idp idp-policy idpengine rulebase-ips rule 1]
   user@host# set then notification log-attacks
   ```

10. Activate the IDP policy.

    ```
    [edit]
    user@host# set security idp active-policy idpengine
    ```

11. Specify a name for the custom attack.

    ```
    [edit security idp]
    user@host# set custom-attack customftp
    ```

12. Set the severity for the custom attack.

    ```
    [edit security idp custom-attack customftp]
    user@host# set severity info
    ```

13. Set the attack type and context for the attack.

    ```
    [edit security idp custom-attack customftp]
    user@host# set attack-type signature context ftp-username
    ```

14. Specify a pattern for the attack.

    ```
    [edit security idp custom-attack customftp]
    user@host# set attack-type signature pattern .*guest.*
    ```

15. Specify a direction for the attack.

    ```
    [edit security idp custom-attack customftp]
    user@host# set attack-type signature direction client-to-server
    ```

16. Specify a name for the custom attack group.

    ```
    [edit security idp]
    user@host# set custom-attack-group cust-group
    ```

17. Specify a list of attacks or attack groups that belongs to the custom attack group.

    ```
    [edit security idp custom-attack-group cust-group]
    user@host# set group-members customftp
    user@host# set group-members ICMP:INFO:TIMESTAMP
    user@host# set group-members "TELNET - Major"
    ```

```
user@host# set group-members dyn1
```

18. Specify a name for the first dynamic attack group.

    ```
    [edit security idp]
    user@host# set dynamic-attack-group dyn1
    ```

19. Configure a filter and set a category value for the filter.

    ```
    [edit security idp dynamic-attack-group dyn1 ]
    user@host# set filters category values TROJAN
    ```

20. Specify a name for the second dynamic attack group.

    ```
    [edit security idp]
    user@host# set dynamic-attack-group dyn2
    ```

21. Configure a filter for the second dynamic attack group and set the direction and its values for this field.

    ```
    [edit security idp dynamic-attack-group dyn2 ]
    user@host# set filters direction expression and
    user@host# set filters direction values server-to-client
    user@host# set filters direction values client-to-server
    ```

22. Specify the trace options and trace file information for the IDP services.

    ```
    [edit]
    user@host# set security idp traceoptions file idpd
    ```

23. Specify the events and other information that needs to be included in the trace output.

    ```
    [edit]
    user@host# set security idp traceoptions flag all
    ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy idpengine {
  rulebase-ips {
    rule 1 {
      match {
        from-zone any;
        source-address any;
        to-zone any;
        destination-address any;
        application default;
        attacks {
```

```
                    custom-attack-groups cust-group;
                    dynamic-attack-groups dyn2;
                }
            }
            then {
                action {
                    no-action;
                }
                notification {
                    log-attacks;
                }
            }
        }
    }
}
active-policy idpengine;
custom-attack customftp {
    severity info;
    attack-type {
        signature {
            context ftp-username;
            pattern .*guest.*;
            direction client-to-server;
        }
    }
}
custom-attack-group cust-group {
    group-members [ customftp ICMP:INFO:TIMESTAMP "TELNET - Major" dyn1 ];
}
dynamic-attack-group dyn1 {
    filters {
        category {
            values TROJAN;
        }
    }
}
dynamic-attack-group dyn2 {
    filters {
        direction {
            expression and;
            values [ server-to-client client-to-server ];
        }
    }
}
traceoptions {
    file idpd;
    flag all;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

NOTE: When you enter **commit** in configuration mode, the configuration is internally verified and then committed. If there are any errors, commit will fail and the errors will be reported.

Verification

Verifying the Configuration

Purpose Verify that the configuration is correct.

Action From operational mode, enter the **show security idp policy-commit-status** command to check the policy compilation or load status.

NOTE: The output of the show security idp policy-commit-status command is dynamic; hence there is no single output for this command.

Verify that the attacks are getting detected as per the configuration, pass traffic through the device which will trigger an attack match. For example, enter the **show security idp status** command to check whether the policy is loaded or not.

user@host> **show security idp status**

```
IDP policy[/var/db/idpd/bins/test.bin.gz.v] and
  detector[/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v]
  loaded successfully.
The loaded policy size is:785 Bytes
```

Enter the **show security idp attack table** command to pass attack traffic and then verify that the attacks are getting detected or not.

NOTE: The command will display the output only when attacks are detected.

user@host> **show security idp attack table**

```
IDP attack statistics:
Attack name #Hits
FTP:USER:ROOT 1
```

Custom Attack Object DFA Expressions

Table 83 on page 178 provides examples of syntax for matching an attack pattern.

Table 83: Example: Custom Attack Object Regular Expressions

Example Syntax	Description	Example Matches
Hello..\B.0.1..00\B...world	There are two aspects to matching: Must match the bitmask pattern: \B.0.0.1..00\B Must match the number of bytes (signified by .) before and after the bitmask pattern.	Matches: Hello..\B.0.11100\B...world Hello..\B.0.10000\B...world Does not match: Hello.\B.0.1..00\B.world Hello..\B.0.1..11\B...world
\X01 86 A5 00 00\X	Pattern with the five specified bytes verbatim.	01 86 A5 00 00
(hello|world)	Pattern with hello or world occurring once.	hello world
(hello|world)+	Pattern with hello or world occurring one or more times.	helloworld worldhello hellohello
\[hello\]	Pattern hello, case insensitive.	hElLo HEllO heLLO
\uHello\u	Pattern hello, Unicode insensitive.	hello 68656c6c6f
hello\sworld	Pattern hello world, the two words separated by a whitespace.	hello world
[c-e]a(d|t)	Pattern with the first letter of c, d, or e; the middle letter a; and ending in d or t.	cat dad eat
[^c-d]a(d|t)	Pattern that begins a letter other than c, d, or e; have the second letter a; and end in d or t.	fad zad
a*b+c	Pattern with any number of a characters (including zero); followed by one or more b characters; followed by a c character.	bc abc aaaabbbbc

Table 83: Example: Custom Attack Object Regular Expressions (continued)

Example Syntax	Description	Example Matches
T[Kk]	Pattern that begins with an uppercase T, followed by a case-insensitive k.	TK Tk
([Tt])k	Pattern that begins with a case-insensitive t, followed by a lowercase k.	Tk Tk
Sea[In]	Pattern that begins with Sea, followed by a lowercase l, m, or n.	Seal Seam Sean
([B-D])at	Pattern that begins with an uppercase B, C, or D, followed by a lowercase at.	Bat Cat Dat
\0133\[hello\]\0135	Pattern that begins with an opening bracket, followed by case-insensitive hello, ending with a closing bracket. This expression uses the \0 expression to signify that the following expression is an octal code, then the octal code for the opening bracket (133) or the closing bracket (135) follows.	[hello] [HeLLo]

Example: Using Pattern Negation

You can use pattern negation to exclude a pattern known to be safe and to match all else.

For example, suppose you are designing an attack object to inspect traffic to an FTP server. You know that account username and passwords are well maintained to ensure that only authorized users can access internal resources. However, as networks grow and new components are added, user accounts can proliferate, thereby increasing network access to specific components. In this example, you have an FTP server on your internal network that has multiple user accounts enabled. To improve security, you want to restrict access to the FTP administrator.

You create an attack object for the FTP service, ftp-username context, and pattern **admin**; and you select the **Negate** check box. The result is an attack object that can flag login attempts by users other than **admin**. You can use this attack object in a rule that logs or drops matching traffic.

See Also

- Creating a Signature Attack Object on page 81
- Creating a Compound Attack Object on page 161

Example: Matching File Extensions

In this example, you want to detect Microsoft Windows metafiles, which use the extensions .emf (Windows Enhanced Metafiles) and .wmf (Microsoft Windows Metafile).

To match either of these file types, use a simple DFA expression:

```
.*\.\[w|emf\]
```

In this expression:

- The period combined with the asterisk (.*) indicates that one or more characters must appear (wildcard match).
- The backslash combined with the period character (\.) indicates that the period character is escaped (the period appears in the pattern).
- The parentheses at the beginning and end of the expression ( ) indicate a group. The pipe character between the e and the w (e|w) indicates an OR relationship between the characters. For this expression, e or w must appear in the pattern to match this expression; only one must be present.
- The opening bracket (\[) indicates the beginning of a case-insensitive match for all characters until the closing bracket (\]) appears.
- The closing bracket (\]) indicates the ending of a case-insensitive match.

See Also

- Creating a Signature Attack Object on page 81
- Creating a Compound Attack Object on page 161

Example: Apache Tomcat Denial-of-Service Attacks

In this example, we assume you have a Web Server running Apache Tomcat. Your security administrator notifies you that a vulnerability has just been announced for Apache Tomcat, and you decide to create a custom attack object to protect your network until you can schedule downtime to patch the server.

The CVE advisory for the vulnerability (http://nvd.nist.gov/nvd.cfm?cvename=CAN-2002-0682) contains the following quotation:

```
A cross-site scripting vulnerability in Apache Tomcat 4.0.3 allows
remote attackers to execute script as other web users via script in a URL with
the /servlet/ mapping, which does not filter the script when an exception is
thrown by the servlet.
```

From this information, you know that the attack uses HTTP. Now you must locate the attack code. The advisory also includes references that link to more information about the attack. Unfortunately, none of the referenced Web pages contain exploit code. After searching the Web using the information you learned from the CVE advisory, you locate some exploit code at http://packetstormsecurity.nl/0210-exploits/neuter.c. Copy the script and move it to the attacker computer in your test lab.

To develop this attack object:

1. Reproduce the attack to determine the attack context, direction, and pattern. Ideally, use **scio ccap** and Wireshark concurrently so you have to run the attack only once.

2. Discover the following elements of the attack signature:

   - Service. You know from the CVE advisory that the attack uses the HTTP protocol. Review the packet capture to confirm the protocol.

   - Context. Use **scio ccap** to determine whether you can match a particular service context. In this example, the signature pattern occurs in the service context HTTP URL Parsed.

   - Pattern. You know from the advisory that the attack occurs using an exploited GET method in the HTTP protocol. Select the frame that contains the GET method to view details for that section of the packet. You can quickly identify the signature pattern as **examples/servlet/AUX**.

   - Direction. Locate the source IP that initiated the session. Because this attack uses TCP, you can use the Follow TCP Stream option in Wireshark to quickly discover the source IP that initiated the session. The attack direction is client-to-server.

3. Create an attack object to match the attack signature. This example uses the following regular expression to match the signature:

   ```
   .*/examples/servlet/AUX|LPT1|CON|PRN.*
   ```

   In this expression:

   - The dot star combination (.*) indicates a wildcard match.

   - The /examples/servlet/ section is taken directly from the packet capture.

   - The parentheses ( ) indicate a group of items, and the pipe character (|) indicates OR. These characters are often used together to indicate that an attack must include one item from the group. In this example, the attack must contain the word aux, lpt1, con, or prn after the string /examples/servlet/.

     Notice that this example uses a group. The packet capture displays the signature pattern as /examples/servlet/AUX. AUX is a Windows device. You have good reason to be on guard for attempts to exploit LPT1, CON, and PRN devices.

4. Test the attack object.

See Also

- Creating a Signature Attack Object on page 81
- Testing a Custom Attack Object on page 80

Listing IDP Test Conditions for a Specific Protocol

When configuring IDP custom attacks, you can specify list test conditions for a specific protocol. To list test conditions for ICMP:

1. List supported test conditions for ICMP and choose the one you want to configure. The supported test conditions are available in the CLI at the **[edit security idp custom-attack test1 attack-type anomaly]** hierarchy level.

   ```
   user@host#set test icmp?

   Possible completions:
    <test>               Protocol anomaly condition to be checked

     ADDRESSMASK_REQUEST
     DIFF_CHECKSUM_IN_RESEND
     DIFF_CHECKSUM_IN_RESPONSE
     DIFF_LENGTH_IN_RESEND
   ```

2. Configure the service for which you want to configure the test condition.

   ```
   user@host# set service ICMP
   ```

3. Configure the test condition (specifying the protocol name is not required).

   ```
   user@host# set test ADDRESSMASK_REQUEST
   ```

4. If you are done configuring the device, enter **commit** from configuration mode.

Understanding IDP Protocol Decoders

Protocol decoders are used by Intrusion Detection and Prevention (IDP) to check protocol integrity and protocol contextual information by looking for anomalies and ensuring that RFC standards are met. An anomaly can be any part of a protocol, such as the header, message body, or other individual fields that deviate from RFC standards for that protocol. For example, in the case of SMTP, if SMTP MAIL TO precedes SMTP HELO, that is an anomaly in the SMTP protocol.

When protocol contextual information is available, protocol decoders check for attacks within those contexts. For example, for SMTP, if an e-mail is sent to user@company.com, user@company.com is the contextual information and SMTP MAIL TO is the context. By using protocol contextual data, rather than the entire packet, for attack detection, protocol decoders improve overall performance and accuracy.

If there is a policy configured with a rule that matches the protocol decoder check for SMTP, the rule triggers and the appropriate action is taken.

The IDP module ships with a preconfigured set of protocol decoders. These protocol decoders have default settings for various protocol-specific contextual checks they perform. You can use these defaults or you can tune them to meet your site's specific needs. To display the list of available protocol decoders, enter the following command:

```
user@host # show security idp sensor-configuration detector protocol-name ?
```

For a more detailed view of the current set of protocol decoders and their default context values, you can view the**detector-capabilities.xml** file located in the **/ar/db/idpd/sec-download** folder on the device. When you download a new security package, you also receive this file which lists current protocols and default decoder context values.

Example: UNIX CDE/dtlogin Vulnerability

In this example, your network includes several user workstations and servers running UNIX. Many UNIX operating systems use the Common Desktop Environment (CDE) as a graphical user interface. Your security administrator notifies you of a new vulnerability in the dtlogin process for CDE (the dtlogin process handles a GUI login process to CDE).

The CERT advisory for the vulnerability (http://www.kb.cert.org/vuls/id/179804) contains the following information:

```
...The dtlogin program contains a "double-free" vulnerability that can be triggered

by a specially crafted X Display Manager Control Protocol (XDMCP) packet... Block
 XDMCP
traffic (177/udp) from untrusted networks such as the Internet...
```

From this information, you know that the attack uses XDMCP protocol packet, and runs on UDP/177. Now you must locate the attack code. The advisory also includes references that link to more information about the attack. One reference, http://lists.immunitysec.com/pipermail/dailydave/2004-March/000402.html, indicates that the person who first reported the attack has also written a script that replicates the attack. Obtain the script and move it to the attacker computer in your test lab.

To develop this attack object:

1. Reproduce the attack to determine the attack context, direction, and pattern. Ideally, use **scio ccap** and Wireshark concurrently so you have to run the attack only once.

2. Discover the elements of the attack signature:

   - Service. You know from the CERT advisory that the attack uses the XDMCP protocol. Review the packet capture in Wireshark to confirm the protocol.
   - Context. Use **scio ccap** to determine whether you can match a particular service context. In this example, the XMCP service contexts are not supported by the IDP system, and the output of **scio ccap** is blank. You must specify the packet context for the attack.
   - Pattern. Using your knowledge of the XDMCP protocol, you identify that the attack uses a non-NUL character (hexadecimal code 00 1b) to specify the connection type, which is invalid (the NUL character represents the Internet connection type in XDMCP). To anchor the non-NUL character in a signature pattern, include some of the preceding bytes as part of the pattern. For this example, you choose to anchor the non-NUL character with the version number (hexadecimal code 00 01) and the request options code (hexadecimal code 00 07). The full attack pattern is 00 01

00 07 followed by five characters of any type, followed by a sixth character and either a non-NUL character (as shown above with 00 1b) or a non-NUL character and another character.

- Direction. Locate the source IP that initiated the session. In this example, you cannot determine the attack direction.

3. Create an attack object to match the attack signature. Use the following regular expression to match the signature:

   ```
   \x00 01 00 07\x.....(.[^\000]|[^\000]..*
   ```

   In this expression:

   - The \x expression indicates a hexadecimal value.
   - The numbers 00 01 00 07 in the XDMP protocol represent the version number (hexadecimal code 00 01 and the request options code (hexadecimal code 00 07).
   - The five periods (.....) indicate five characters of any kind.
   - The parentheses ( ) indicates a group of items, and the pipe character (|) indicates OR. These characters are often used together to indicate that an attack must include one item from the group.
   - The opening and closing brackets combined with a caret [^ indicates negation.
   - The backslash combined with a zero (\0) indicates an octal code number.
   - The 00 characters are hexadecimal code for a NUL character. In this example, the attack must contain a non-NUL character, either preceded or followed by another character ([^\000] or [^\000]).
   - The dot star combination (.*) indicates a wildcard match. When used at the end of an expression, the wildcard indicates that anything can follow the specified expression.

4. Test the attack object.

See Also

- Creating a Signature Attack Object on page 81
- Testing a Custom Attack Object on page 80

Example: Detecting a Worm

Worms and Trojans often bypass firewalls and other traditional security measures to enter a network. In this example, you create a custom attack object to detect the Blaster worm on your network.

The CERT advisory (http://www.cert.org/advisories/CA-2003-20.html) for the Blaster worm gives the following information:

```
The W32/Blaster worm exploits a vulnerability in Microsoft's DCOM RPC interface..."
```

From this information, you know that the attack uses DCOM exploit, a previously identified security hole. Now you must locate the attack code. The advisory also includes references that link to more information about the attack. Unfortunately, none of the referenced Web pages contain exploit code. After searching the Web using the information you learned from the CERT advisory, you locate exploit code on PacketStorm (http://packetstormsecurity.com/0307-exploits/dcom.c).

To develop this attack object:

1. Reproduce the attack to determine the attack context, direction, and pattern. Ideally, use **scio ccap** and Wireshark concurrently so you have to run the attack only once.

2. Discover the elements of the attack signature:

   - Service. You know from the CERT advisory that the attack uses ICMP, for which the IDP OS does not support service contexts. Review the packet capture to confirm the protocol as ICMP.
   - Context. Use **scio ccap** to determine whether we can match a particular service context. In this example, the ICMP service contexts are not supported by the IDP system, and the output of **scio ccap** is blank. You must specify the first packet context for the attack.
   - Pattern. Select the first frame listed in Wireshark and review the information in the second section. Because you know that ICMP packets should not contain data, you investigate the 64 byte data payload. You can easily see the irregular payload is multiple “AA” characters, which is probably code attempting to overflow a buffer. Because this pattern is unusual in the context of an ICMP packet, select it as your signature.
   - Direction. Locate the source IP that initiated the session. In this example, you cannot determine the attack direction.

3. Create an attack object to match the attack signature. In this example, we use the following regular expression to match the signature:

   ```
   \X AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA  AA AA AA AA AA AA AA
   AA AA AA AA AA AA AA AA  AA AA AA AA AA AA AA AA AA AA AA  AA AA AA AA AA AA
   AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA \X.*
   ```

   In this expression:

   - The \X expression indicates that a hexadecimal value is to follow.
   - The dot star combination (.*) indicates a wildcard match. When used at the end of an expression, the wildcard indicates that anything can follow the specified expression.

4. Test the attack object.

See Also

- Creating a Signature Attack Object on page 81
- Testing a Custom Attack Object on page 80

Example: Compound Signature to Detect Exploitation of an HTTP Vulnerability

Some attacks are crafted to appear benign when viewed at a packet-by-packet level. For these attacks, you can create a compound signature that detects multiple signature patterns in multiple contexts (service, nonservice, or both).

In this example, you have a Web server that uses Microsoft FrontPage Server extensions. Your security administrator notifies you of a new buffer overflow vulnerability in the FrontPage Server extensions.

The BugTraq advisory for the vulnerability (http://www.securityfocus.com/bid/9007/discussion/) contains the following information:

```
Microsoft FrontPage Server Extensions are prone to a remotely exploitable
buffer overrun vulnerability ... It is possible to trigger this condition with a

chunked-encoded HTTP POST request...
```

The following proof-of-concept example is also provided:

```
POST /_vti_bin/_vti_aut/fp30reg.dll HTTP/1.1
Transfer-Encoding: chunked
PostLength
PostData
0
```

Additionally, a link to the compiled exploit is included.

From this information, you know that the attack uses the HTTP protocol and that at least part of the attack uses the POST method. Use the link to the compiled exploit to obtain the script, and move it to the attacker computer in your test lab.

To develop this attack object:

1. Reproduce the attack to determine the attack context, direction, and pattern. Ideally, use **scio ccap** and Wireshark concurrently so you only have to run the attack only once.

2. Discover the elements of the attack signature:

   - Service. You know from the BugTraq advisory that the attack uses the HTTP protocol. Review the packet capture and locate the HTTP protocol usage.
   - Context. Use **scio ccap** to determine whether you can match a particular service context. In this example, the service context is HTTP URL Parsed.
   - Pattern. You quickly identify the signature pattern POST /_vti_bin/_vti_aut/fp30reg.dll within the HTTP service.

     However, because this pattern might trigger false positives, you also determine a second signature pattern to ensure that your rule detects only the attack. In this case, the second signature (noted in the BugTraq advisory) is **Transfer-Encoding: chunked**.

- Direction. Locate the source IP that initiated the session. In this example, the attack direction for both signature patterns is client-to-server.

3. Create an attack object to match the attack signature. Use the following regular expression to match the first signature:

   ```
   \[/_vti_bin/_vti_aut/fp30reg\.dll\].*
   ```

   In this expression:

   - The opening bracket (\[) indicates the beginning of a case-insensitive match for all characters until the closing bracket appears.
   - The pattern /_vti_bin/_vti_aut/fp30reg is a direct character match.
   - The backslash combined with the period (\.) indicates that the period is escaped (the period appears in the pattern).
   - The closing bracket (\]) indicates the end of a case-insensitive match.
   - The period combined with the asterisk character (.*) indicates that one or more characters must appear.

4. Add a second signature. Use the following regular expression to match the second signature:

   ```
   \[Transfer-Encoding: +chunked\]
   ```

   In this expression:

   - The opening bracket (\[) indicates the beginning of a case-insensitive match for all characters until the closing bracket appears.
   - The pattern Transfer-Encoding: is a direct character match.
   - The plus sign (+) indicates that a space character must appear one or more times within the pattern.
   - The pattern chunked is a direct character match.
   - The closing bracket (\]) indicates the end of a case-insensitive match.

5. Test the attack object.

See Also

- Creating a Signature Attack Object on page 81
- Testing a Custom Attack Object on page 80

Example: Using Time Binding Parameters to Detect a Brute Force Attack

The time binding constraint requires the pattern to occur a certain number of times within a minute in order for the traffic to be considered a match.

You can use the time binding parameter along with the signature to detect signs of a brute force attack. A user changing her password is a harmless event, and is normally seen occasionally on the network. However, thousands of password changes in a minute is suspicious.

In a brute force attack, the attacker attempts to break through system defenses using sheer force, typically by overwhelming the destination server capacity or by repeated, trial-and-error attempts to match authentication credentials. In a brute force login attack, the attackers first gather a list of usernames and a password dictionary. Next, the attacker uses a tool that enters the first password in dictionary for the first user in the list, then tries every password for every user until it gets a match. If the attacker tries every combination of usernames and passwords, they always succeed. However, brute force attacks often fail because the password dictionary is typically limited (does not contain all possible passwords) and the attack tool does not perform permutations on the password (such as reversing letters or changing case).

In this example, you create a signature attack object that detects an excessive number of password changes for users authenticated via HTTP (a Web-based application).

First, you configure an attack pattern:

```
.*/\[changepassword\.cgi\]
```

In this expression:

- The dot star combination (.*) indicates a wildcard match.
- The backslash before a character indicates that the character represents a regular expression and must be escaped. In this case, the character is an opening bracket. The backslash is also used in this expression before the file extension marker (the dot) and before the closing bracket.
- The name of the cgi script that is used to change user passwords is included, as well as the cgi extension.
- For context, select **HTTP-URL-PARSED** from the list because you are attempting to detect password changes that occur for Web-based applications. The changepassword.cgi script, when used, appears as part of the URL, but you need to tell the IDP Series device to parse the URL in order to find the name.

Next, you configure time binding.

In these settings:

- Scope is set to **Peer** so the attack pattern can match the event regardless of source or destination.
- Count is set to high number (to 1000) to avoid false positives. This value means that the changepassword.cgi script must appear in a URL 1000 times before the attack object is matched.

See Also

- Creating a Signature Attack Object on page 81

- Creating a Compound Attack Object on page 161
- Testing a Custom Attack Object on page 80

Reference: Custom Attack Object Protocol Numbers

Table 84 on page 189 protocol numbers used in the IDP system.

Table 84: IDP Attack Objects: Protocol Numbers

Protocol Name	Protocol Number
HOPOPT	0
ICMP	1
IGMP	2
GGP	3
IPIP	4
ST	5
TCP	6
CBT	7
EGP	8
IGP	9
BBN-RCC-MON	10
NVP-II	11
PUP	12
ARGUS	13
EMCON	14
XNET	15
CHAOS	16
UDP	17
MUX	18
DCN-MEAS	19

Table 84: IDP Attack Objects: Protocol Numbers *(continued)*

Protocol Name	Protocol Number
HMP	20
PRM	21
XND-IDP	22
TRUNK-1	23
TRUNK-2	24
LEAF-1	25
LEAF-2	26
RDP	27
IRTP	28
ISO-TP4	29
NETBLT	30
MFE-NSP	31
MERIT-INP	32
SEP	33
3PC	34
IDPR	35
XTP	36
DDP	37
TP_PLUS_PLUS	39
IL	40
IPV6	41
SDRP	42
IPV6-ROUTING	43
IDV6-FRAGMENT	44

Table 84: IDP Attack Objects: Protocol Numbers (continued)

Protocol Name	Protocol Number
IDRP	45
RSVP	46
GRE	47
MHRP	48
BNA	49
ESP	50
AH	51
I-NLSP	52
SWIPE	53
NARP	54
MOBILE	55
TLSP	56
SKIP	57
IPV6-ICMP	58
IPV6-NONXT	59
IPV6-OPTS	60
AHIP	61
CFTP	62
ALNP	63
SAT-EXPAK	64
KRYPTOLAN	65
RVD	66
IPPC	67
ADFSP	68

Table 84: IDP Attack Objects: Protocol Numbers *(continued)*

Protocol Name	Protocol Number
SAT-MON	69
VISA	70
IPCV	71
CPNX	72
CPHB	73
WSN	74
PVP	75
BR-SAT-MON	76
SUN-ND	77
WB-MON	78
WB-EXPAK	79
ISO-IP	80
VMTP	81
SECURE-VMTP	82
VINES	83
TTP	84
NSFNET-IBP	85
DGP	86
TCF	87
EIGRP	88
OSPFIGP	89
SPRITE-RPC	90
LARP	91
MTP	92

Table 84: IDP Attack Objects: Protocol Numbers (continued)

Protocol Name	Protocol Number
AX_25	93
IPIP	94
MICP	95
SCC-SP	96
ETHERIP	97
ENCAP	98
APES	99
GMTP	100
IFMP	101
PNNI	102
PIM	103
ARIS	104
SCPS	105
QNX	106
A/N	107
IPCOMP	108
SNP	109
COMPAT-PEER	110
IPZ-IN-IP	111
VRRP	112
PGM	113
HOP-O	114
L2TP	115
DDX	116

Table 84: IDP Attack Objects: Protocol Numbers (continued)

Protocol Name	Protocol Number
IATP	117
STP	118
SRP	119
UTI	120
SMP	121
SSM	122
PTP	123
ISIS	124
FIRE	125
CRTP	126
CRUDP	127
SSCOPMCE	128
IPLT	129
SPS	130
PIPE	131
SCTP	132
FC	133
RSVP-E2E-IGNORE	134
n/a	
n/a	
n/a	
RESERVED	255

Reference: Nonprintable and Printable ASCII Characters

The following tables provide details on ASCII representation of nonprintable and printable characters.

Table 85: ASCII Reference: Nonprintable Characters

Dec	Hex	Oct	Char	Comment
0	0	000	NUL	Null
1	1	001	SOH	Start of Heading
2	2	002	STX	Start of Text
3	3	003	ETX	End of Text
4	4	004	EOT	End of Transmission
5	5	005	ENQ	Enquiry
6	6	006	ACK	Acknowledge
7	7	007	BEL	Bell
8	8	010	BS	Backspace
9	9	011	TAB	Horizontal Tab
10	A	012	LF	Line Feed
11	B	013	VT	Vertical Tab
12	C	014	FF	Form Feed
13	D	015	CR	Carriage Return
14	E	016	SO	Shift Out
15	F	017	SI	Shift In
16	10	020	DLE	Data Link Escape
17	11	021	DC1	Device Control 1
18	12	022	DC2	Device Control 2
19	13	023	DC3	Device Control 3
20	14	024	DC4	Device Control 4

Table 85: ASCII Reference: Nonprintable Characters (continued)

Dec	Hex	Oct	Char	Comment
21	15	025	NAK	Negative Acknowledgement
22	16	026	SYN	Synchronous Idle
23	17	027	ETB	End of Transmission Block
24	18	030	CAN	Cancel
25	19	031	EM	End of Medium
26	1A	032	SUB	Substitute
27	1B	033	ESC	Escape
28	1C	034	FS	File Separator
29	1D	035	GS	Group Separator
30	1E	036	RS	Record Separator
31	1F	037	US	Unit Separator

Table 86: ASCII Reference: Printable Characters

Dec	Hex	Oct	Char
32	20	040	Space
33	21	041	!
34	22	042	
35	23	043	#
36	24	044	$
37	25	045	%
38	26	046	&
39	27	047	
40	28	050	(

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
41	29	051	)
42	2A	052	*
43	2B	053	+
44	2C	054	,
45	2D	055	-
46	2E	056	.
47	2F	057	/
48	30	060	0
49	31	061	1
50	32	062	2
51	33	063	3
52	34	064	4
53	35	065	5
54	36	066	6
55	37	067	7
56	38	070	8
57	39	071	9
58	3A	072	:
59	3B	073	;
60	3C	074	<
61	3D	075	=
62	3E	076	>
63	3F	077	?
64	40	100	@

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
65	41	101	A
66	42	102	B
67	43	103	C
68	44	104	D
69	45	105	E
70	46	106	F
71	47	107	G
72	48	110	H
73	49	111	I
74	4A	112	J
75	4B	113	K
76	4C	114	L
77	4D	115	M
78	4E	116	N
79	4F	117	O
80	50	120	P
81	51	121	Q
82	52	122	R
83	53	123	S
'84	54	124	T
85	55	125	U
86	56	126	V
87	57	127	W
88	58	130	X

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
89	59	131	Y
90	5A	132	Z
91	5B	133	[
92	5C	134	\
93	5D	135	]
94	5E	136	^
95	5F	137	_
96	60	140	`
97	61	141	a
98	62	142	b
99	63	143	c
100	64	144	d
101	65	145	e
102	66	146	f
103	67	147	g
104	68	150	h
105	69	151	I
106	6A	152	j
107	6B	153	k
108	6C	154	l
109	6D	155	m
110	6E	156	n
111	6F	157	o
112	70	160	p

Table 86: ASCII Reference: Printable Characters *(continued)*

Dec	Hex	Oct	Char
113	71	161	q
114	72	162	r
115	73	163	s
116	74	164	t
117	75	165	u
118	76	166	v
119	77	167	w
120	78	170	x
121	79	171	y
122	7A	172	z
123	7B	173	{
124	7C	174	|
125	7D	175	}
126	7E	176	~
127	7F	177	DEL
128	80	200	Ç
129	81	201	ü
130	82	202	é
131	83	203	â
132	84	204	ä
133	85	205	à
134	86	206	å
135	87	207	ç
136	88	210	ê

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
137	89	211	ë
138	8A	212	è
139	8B	213	ï
140	8C	214	î
141	8D	215	ì
142	8E	216	Ä
143	8F	217	Å
144	90	220	É
145	91	221	æ
146	92	222	Æ
147	93	223	ô
148	94	224	ö
149	95	225	ò
150	96	226	û
151	97	227	ù
152	98	230	ÿ
153	99	231	Ö
154	9A	232	Ü
155	9B	233	¢
156	9C	234	£
157	9D	235	¥
158	9E	236	P
159	9F	237	ƒ
160	A0	240	á

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
161	A1	241	í
162	A2	242	ó
163	A3	243	ú
164	A4	244	ñ
165	A5	245	Ñ
166	A6	246	ª
167	A7	247	º
168	A8	250	¿
169	A9	251	◎
170	AA	252	
171	AB	253	½
172	AC	254	¼
173	AD	255	¡
174	AE	256	"
175	AF	257	"
176	B0	260	¦
177	B1	262	¦
178	B2	262	¦
179	B3	263	¦
180	B4	264	¦
181	B5	265	¦
182	B6	266	¦
183	B7	267	+
184	B8	270	+

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
185	B9	271	¦
186	BA	272	¦
187	BB	273	+
188	BC	274	+
189	BD	275	+
190	BE	276	+
191	BF	277	+
192	C0	300	+
193	C1	301	-
194	C2	302	-
195	C3	303	+
196	C4	304	-
197	C5	305	+
198	C6	306	¦
199	C7	307	¦
200	C8	310	+
201	C9	311	+
202	CA	312	-
203	CB	313	-
204	CC	314	¦
205	CD	315	-
206	CE	316	+
207	CF	317	-
208	D0	320	-

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
209	D1	321	-
210	D2	322	-
211	D3	323	+
212	D4	324	+
213	D5	325	+
214	D6	326	+
215	D7	327	+
216	D8	330	+
217	D9	331	+
218	DA	332	+
219	DB	333	¦
220	DC	334	_
221	DD	335	¦
222	DE	336	¦
223	DF	337	¯
224	E0	340	a
225	E1	341	ß
226	E2	342	G
227	E3	343	p
228	E4	344	S
229	E5	345	s
230	E6	346	µ
231	E7	347	t
232	E8	350	F

Table 86: ASCII Reference: Printable Characters (continued)

Dec	Hex	Oct	Char
233	E9	351	T
234	EA	352	O
235	EB	353	d
236	EC	354	8
237	ED	355	f
238	EE	356	e
239	EF	357	n
240	F0	360	=
241	F1	361	+/-
242	F2	362	=
243	F3	363	=
244	F4	364	(
245	F5	365	)
246	F6	366	÷
247	F7	367	˜
248	F8	370	o
249	F9	371	
250	FA	372	
251	FB	373	v
252	FC	374	n
253	FD	375	^{2}
254	FE	376	¦
255	FF	377	

Example: Configuring IDP Protocol Decoders

This example shows how to configure IDP protocol decoder tunables.

Requirements

Before you begin, review the IDP protocol decoders feature. See “Understanding IDP Protocol Decoders” on page 182.

Overview

The Junos IDP module ships with a set of preconfigured protocol decoders. These protocol decoders have default settings for various protocol-specific contextual checks that they perform. You can use the default settings or tune them to meet your site's specific needs. This example shows you how to tune the protocol decoder for FTP.

Configuration

Step-by-Step Procedure

To configure IDP protocol decoder tunables:

1. View the list of protocols that have tunable parameters.

```
[edit]
user@host# edit security idp sensor-configuration detector protocol-name FTP
```

2. Configure tunable parameters for the FTP protocol.

```
[edit security idp sensor-configuration-detector protocol-name FTP]
user@host# set tunable-name sc_ftp_failed_logins tunable-value 4
user@host# set tunable-name sc_ftp_failed_flags tunable value 1
user@host# set tunable-name sc_ftp_line_length tunable-value 1024
user@host# set tunable-name sc_ftp_password_length tunable-value 64
user@host# set tunable-name sc_ftp_sitestring_length tunable-value 512
user@host# set tunable-name sc_ftp_username_length tunable-value 32
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

Verification

To verify the configuration is working properly, enter the **show security idp status** command.

Understanding Multiple IDP Detector Support

When a new security package is received, it contains attack definitions and a detector. In any given version of a security package, the attack definitions correspond to the capabilities of the included detector. When policy aging is disabled on the device (see the reset-on-policy statement for policy aging commands), only one policy is in effect at any given time. But if policy aging is enabled and there is a policy update, the existing policy is not unloaded when the new policy is loaded. Therefore, both policies can be in effect on the device. In this case, all existing sessions will continue to be inspected by existing policies and new sessions are inspected with new policies. Once all the existing sessions using the older policy have terminated or expired, the older policy is then unloaded.

When a policy is loaded, it is also associated with a detector. If the new policy being loaded has an associated detector that matches the detector already in use by the existing policy, the new detector is not loaded and both policies use a single associated detector. But if the new detector does not match the current detector, the new detector is loaded along with the new policy. In this case, each loaded policy will then use its own associated detector for attack detection.

Note that a maximum of two detectors can be loaded at any given time. If two detectors are already loaded (by two or more policies), and loading a new policy requires also loading a new detector, then at least one of the loaded detectors must be unloaded before the new detector can be loaded. Before a detector is unloaded, all policies that use the corresponding detector are unloaded as well.

You can view the current policy and corresponding detector version by entering the following command:

```
user@host> show security idp status
```

Understanding Content Decompression

In application protocols like HTTP, the content could be compressed and then transmitted over the network. The patterns will not match the compressed content, because the signature patterns are written to match the unencoded traffic data. In this case IDP detection is evaded. To avoid IDP detection evasion on the HTTP compressed content, an IDP submodule has been added that decompresses the protocol content. The signature pattern matching is done on the decompressed content.

To display the status of all IPS counter values, enter the following command:

```
user@host> show security idp counters ips
```

Some attacks are introduced through compressed content. When the content is decompressed, it can inflate to a very large size taking up valuable system resources resulting in denial of service. This type of attack can be recognized by the ratio of decompressed data size to compressed data size. The content-decompress-ratio-over-limit counter identifies the number of incidents where this ratio has been exceeded. The default ratio is considered consistent with a typical environment. In some cases, however, this ratio might need to be adjusted by resetting the **content-decompress-ratio-over-limit** value. Keep in mind, however, that a higher ratio lessens the chance of detecting this type of attack.

The content-decompress-memory-over-limit counter identifies the number of incidents where the amount of decompressed data exceeded the allocated memory. The default memory allocation provides 33 KB per session for an average number of sessions requiring decompression at the same time. To determine if this value is consistent with your environment, analyze values from decompression-related counters and the total number of IDP sessions traversing the device, and estimate the number of sessions requiring decompression at the same time. Assuming that each of these sessions requires 33 KB of memory for decompression, compare your estimated needs to the default value. If necessary, you can adjust the memory allocation by resetting the **content-decompression-max-memory-kb** value. Note that because content decompression requires a significant allocation of memory, system performance will be impacted by increasing the maximum memory allocation for decompression.

Example: Configuring IDP Content Decompression

This example shows how to configure IDP content decompression.

Requirements

Before you begin, review the IDP content decompression feature. See “Understanding Content Decompression” on page 207

Overview

The decompression feature is disabled by default. In this example, you enable the detector, configure the maximum memory to 50,000 kilobytes, and configure a maximum decompression ratio of 16:1.

NOTE: **Enabling decompression will result in a reduction in performance on your device.**

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp sensor-configuration detector protocol-name HTTP tunable-name
   sc_http_jpeg_depth tunable-value 0
set security idp sensor-configuration ips content-decompression-max-memory-kb 5000
set security idp sensor-configuration ips content-decompression-max-ratio 16
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see Using the CLI Editor in Configuration Mode in the CLI User Guide.

To configure IDP content decompression:

1. Enable the detector.

   ```
   [edit]
   user@host# set security idp sensor-configuration detector protocol-name HTTP
     tunable-name sc_http_jpeg_depth tunable-value 0
   ```

   NOTE: To disable the detector, set the **tunable-value** to 0.

2. Configure the maximum memory in kilobytes.

   ```
   [edit]
   user@host# set security idp sensor-configuration ips
     content-decompression-max-memory-kb 5000
   ```

3. Configure the maximum decompression ratio.

   ```
   [edit]
   user@host# set security idp sensor-configuration ips
     content-decompression-max-ratio 16
   ```

Results

From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
sensor-configuration {
  ips {
    content-decompression-max-memory-kb 5000;
    content-decompression-max-ratio 16;
  }
  detector {
    protocol-name HTTP {
      tunable-name sc_http_jpeg_depth {
        tunable-value 0;
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verify the content-decompress counters on page 210

Verify the content-decompress counters

Purpose Verify the content-decompress counters. The content-decompress counters provide statistics on decompression processing.

Action From operational mode, enter the **show security idp status** command.

```
State of IDP: Default,  Up since: 2018-04-12 04:32:32 PDT (00:26:54 ago)

Packets/second: 0               Peak: 0 @ 2018-04-12 04:32:32 PDT
KBits/second  : 0               Peak: 0 @ 2018-04-12 04:32:32 PDT
Latency (microseconds): [min: 0] [max: 0] [avg: 0]

Packet Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2018-04-12 04:32:32 PDT]
  TCP: [Current: 0] [Max: 0 @ 2018-04-12 04:32:32 PDT]
  UDP: [Current: 0] [Max: 0 @ 2018-04-12 04:32:32 PDT]
  Other: [Current: 0] [Max: 0 @ 2018-04-12 04:32:32 PDT]

Session Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]
  Policy Name : none
```

Meaning The output provides the status of the current IDP policy.

Understanding IDP Signature-Based Attacks

To configure a custom attack object, you specify a unique name for it and then specify additional information, which can make it easier for you to locate and maintain the attack object.

Certain properties in the attack object definitions are common to all types of attacks, such as attack name, severity level, service or application binding, time binding, and protocol or port binding. Some fields are specific to an attack type and are available only for that specific attack definition.

Signature attack objects use a stateful attack signature (a pattern that always exists within a specific section of the attack) to detect known attacks. They also include the protocol or service used to perpetrate the attack and the context in which the attack occurs. The following properties are specific to signature attacks, and you can configure them when configuring signature attack—attack context, attack direction, attack pattern, and protocol-specific parameters (TCP, UDP, ICMP, or IP header fields).

When configuring signature-based attacks, keep the following in mind:

- Attack context and direction are mandatory fields for the signature attack definition.
- Pattern negation is supported for packet, line, and application-based contexts only and not for stream and normalized stream contexts.
- When configuring the protocol-specific parameters, you can specify fields for only one of the following protocols—IP, TCP, UDP, or ICMP.
- When configuring a protocol binding, you can specify only one of the following—IP, ICMP, TCP, UDP, RPC or applications.
  - IP—Protocol number is a mandatory field.
  - TCP and UDP—You can specify either a single port (**minimum-port**) or a port range (**minimum-port** and **maximum-port**). If you do not specify a port, the default value is taken (**0-65535**).
  - RPC—Program number is a mandatory field.

Example: Configuring IDP Signature-Based Attacks

This example shows how to create a signature-based attack object.

Requirements

Before you begin, configure network interfaces.

Overview

In this example, you create a signature attack called sig1 and assign it the following properties:

- Recommended action (drop packet)—Drops a matching packet before it can reach its destination but does not close the connection.
- Time binding—Specifies the scope as **source** and the count as **10**. When scope is **source**, all attacks from the same source are counted, and when the number of attacks reaches the specified count (**10**), the attack is logged. In this example, every tenth attack from the same source is logged.
- Attack context (packet)—Matches the attack pattern within a packet.
- Attack direction (any)—Detects the attack in both directions—client-to-server and server-to-client traffic.
- Protocol (TCP)—Specifies the TTL value of 128.

- Shellcode (Intel)—Sets the flag to detect shellcode for Intel platforms.
- Protocol binding—Specifies the TCP protocol and ports 50 through 100.

Once you have configured a signature-based attack object, you specify the attack as match criteria in an IDP policy rule. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp custom-attack sig1 severity major
set security idp custom-attack sig1 recommended-action drop-packet
set security idp custom-attack sig1 time-binding scope source count 10
set security idp custom-attack sig1 attack-type signature context packet
set security idp custom-attack sig1 attack-type signature shellcode intel
set security idp custom-attack sig1 attack-type signature protocol ip ttl value 128 match
  equal
set security idp custom-attack sig1 attack-type signature protocol-binding tcp
  minimum-port 50 maximum-port 100
set security idp custom-attack sig1 attack-type signature direction any
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To create a signature-based attack object:

1. Specify a name for the attack.

```
[edit]
user@host# edit security idp custom-attack sig1
```

2. Specify common properties for the attack.

```
[edit security idp custom-attack sig1]
user@host# set severity major
user@host# set recommended-action drop-packet
user@host# set time-binding scope source count 10
```

3. Specify the attack type and context.

```
[edit security idp custom-attack sig1]
user@host# set attack-type signature context packet
```

4. Specify the attack direction and the shellcode flag.

```
[edit security idp custom-attack sig1]
user@host# set attack-type signature shellcode intel
```

5. Set the protocol and its fields.

   ```
   [edit security idp custom-attack sig1]
   user@host# set attack-type signature protocol ip ttl value 128 match equal
   ```

6. Specify the protocol binding and ports.

   ```
   [edit security idp custom-attack sig1]
   user@host# set attack-type signature protocol-binding tcp minimum-port 50
     maximum-port 100
   ```

7. Specify the direction.

   ```
   [edit security idp custom-attack sig1]
   user@host# set attack-type signature direction any
   ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
custom-attack sig1 {
  recommended-action drop-packet;
  severity major;
  time-binding {
    count 10;
    scope source;
  }
  attack-type {
    signature {
      protocol-binding {
        tcp {
          minimum-port 50 maximum-port 100;
        }
      }
      context packet;
      direction any;
      shellcode intel;
      protocol {
        ip {
          ttl {
            match equal;
            value 128;
          }
        }
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verifying the Configuration on page 214

Verifying the Configuration

Purpose Verify that the signature-based attack object was created.

Action From operational mode, enter the **show security idp status** command.

Understanding IDP Protocol Anomaly-Based Attacks

A protocol anomaly attack object detects unknown or sophisticated attacks that violate protocol specifications (RFCs and common RFC extensions). You cannot create new protocol anomalies, but you can configure a new attack object that controls how your device handles a predefined protocol anomaly when detected.

The following properties are specific to protocol anomaly attacks:

- Attack direction
- Test condition

When configuring protocol anomaly-based attacks, keep the following in mind:

- The service or application binding is a mandatory field for protocol anomaly attacks. Besides the supported applications, services also include IP, TCP, UDP, ICMP, and RPC.
- The attack direction and test condition properties are mandatory fields for configuring anomaly attack definitions.

Example: Configuring IDP Protocol Anomaly-Based Attacks

This example shows how to create a protocol anomaly-based attack object.

- Requirements on page 214
- Overview on page 214
- Configuration on page 215
- Verification on page 216

Requirements

Before you begin, configure network interfaces.

Overview

In this example, you create a protocol anomaly attack called anomaly1 and assign it the following properties:

- Time binding—Specifies the scope as **peer** and count as **2** to detect anomalies between source and destination IP addresses of the sessions for the specified number of times.
- Severity (info)—Provides information about any attack that matches the conditions.
- Attack direction (any)—Detects the attack in both directions—client-to-server and server-to-client traffic.
- Service (TCP)—Matches attacks using the TCP service.
- Test condition (OPTIONS_UNSUPPORTED)—Matches certain predefined test conditions. In this example, the condition is to match if the attack includes unsupported options.
- Shellcode (sparc)—Sets the flag to detect shellcode for Sparc platforms.

Once you have configured the protocol anomaly-based attack object, you specify the attack as match criteria in an IDP policy rule. See "Example: Defining Rules for an IDP IPS RuleBase" on page 66.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp custom-attack anomaly1 severity info
set security idp custom-attack anomaly1 time-binding scope peer count 2
set security idp custom-attack anomaly1 attack-type anomaly test
  OPTIONS_UNSUPPORTED
set security idp custom-attack sa
set security idp custom-attack sa attack-type anomaly service TCP
set security idp custom-attack sa attack-type anomaly direction any
set security idp custom-attack sa attack-type anomaly shellcode sparc
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To create a protocol anomaly-based attack object:

1. Specify a name for the attack.

```
[edit]
user@host# edit security idp custom-attack anomaly1
```

2. Specify common properties for the attack.

```
[edit security idp custom-attack anomaly1]
user@host# set severity info
user@host# set time-binding scope peer count 2
```

3. Specify the attack type and test condition.

```
[edit security idp custom-attack anomaly1]
user@host# set attack-type anomaly test OPTIONS_UNSUPPORTED
```

4. Specify other properties for the anomaly attack.

   ```
   [edit security idp custom-attack anomaly1]
   user@host# set attack-type anomaly service TCP
   user@host# set attack-type anomaly direction any
   user@host# attack-type anomaly shellcode sparc
   ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
custom-attack anomaly1 {
    severity info;
    time-binding {
        count 2;
        scope peer;
    }
    attack-type {
        anomaly {
            test OPTIONS_UNSUPPORTED;
            service TCP;
            direction any;
            shellcode sparc;
        }
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 216

Verifying the Configuration

Purpose Verify that the protocol anomaly-based attack object was created.

Action From operational mode, enter the **show security idp status** command.

IDP Extended Package Configuration Overview

The Junos OS Intrusion Detection and Prevention (IDP) policy enables you to selectively enforce various attack detection and prevention techniques on network traffic passing through an IDP-enabled device. It allows you to define policy rules to match a section of traffic based on a zone, network, and application, and then take active or passive preventive actions on that traffic.

An IDP policy defines how your device handles the network traffic. It allows you to enforce various attack detection and prevention techniques on traffic traversing your network.

A policy is made up of rule bases, and each rule base contains a set of rules. You define rule parameters, such as traffic match conditions, action, and logging requirements, then add the rules to rule bases. After you create an IDP policy by adding rules in one or more rule bases, you can select that policy to be the active policy on your device.

To configure the IDP extended package (IPS-EP) perform the following steps:

1. Enable IPS in a security policy. See "Example: Enabling IDP in a Security Policy" on page 45.

2. Configure IDP policy rules, IDP rule bases, and IDP rule actions. See "Example: Inserting a Rule in the IDP Rulebase" on page 63, "Example: Defining Rules for an IDP IPS RuleBase" on page 66, and "Example: Configuring and Applying Rewrite Rules on a Security Device" on page 238 topics.

3. Configure IDP custom signatures. See "Understanding IDP Signature-Based Attacks" on page 210 and "Example: Configuring IDP Signature-Based Attacks" on page 211 topics.

4. Update the IDP signature database. See "Updating the IDP Signature Database Overview" on page 32.

Release History Table

Release	Description
15.1X49-D140	Starting with Junos OS Release 15.1X49-D140, the maximum number of characters allowed for a custom attack object name is 60. You can validate the statement using the **set security idp custom-attack** command.

Related Documentation

- IDP Policy Rules and IDP Rule Bases on page 55
- IDP Signature Database Overview on page 31

Applications and Application Sets for IDP Policies

Supported Platforms SRX Series, vSRX

Applications or services represent Application Layer protocols that define how data is structured as it travels across the network.

For more information, see the following topics:

- Understanding IDP Application Sets on page 218
- Example: Configuring IDP Applications Sets on page 218
- Example: Configuring IDP Applications and Services on page 221

Understanding IDP Application Sets

Applications or services represent Application Layer protocols that define how data is structured as it travels across the network. Because the services you support on your network are the same services that attackers must use to attack your network, you can specify which services are supported by the destination IP to make your rules more efficient. Juniper Networks provides predefined applications and application sets that are based on industry-standard applications. If you need to add applications that are not included in the predefined applications, you can create custom applications or modify predefined applications to suit your needs.

You specify an application, or service, to indicate that a policy applies to traffic of that type. Sometimes the same applications or a subset of them can be present in multiple policies, making them difficult to manage. Junos OS allows you to create groups of applications called *application set*.

Application sets simplify the process by allowing you to manage a small number of application sets, rather than a large number of individual application entries.

The application (or application set) is configured as a match criterion for packets. Packets must be of the application type specified in the policy for the policy to apply to the packet. If the packet matches the application type specified by the policy and all other criteria match, then the policy action is applied to the packet. You can use predefined or custom applications and refer to them in a policy.

See Also

- Example: Configuring IDP Applications and Services on page 221

Example: Configuring IDP Applications Sets

This example shows how to create an application set and associate it with an IDP policy.

- Requirements on page 218
- Overview on page 219
- Configuration on page 219
- Verification on page 220

Requirements

Before you begin:

- Configure network interfaces.

- Enable IDP application services in a security policy. See "Example: Enabling IDP in a Security Policy" on page 45.
- Define applications. See *Example: Configuring Applications and Application Sets*.

Overview

To configure an application set, you add predefined or custom applications separately to an application set and assign a meaningful name to the application set. Once you name the application set you specify the name as part of the policy. For this policy to apply on a packet, the packet must match any one of the applications included in this set.

This example describes how to create an application set called SrvAccessAppSet and associate it with IDP policy ABC. The application set SrvAccessAppSet combines three applications. Instead of specifying three applications in the policy rule, you specify one application set. If all of the other criteria match, any one of the applications in the application set serves as valid matching criteria.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set applications application-set SrvAccessAppSet application junos-ssh
set applications application-set SrvAccessAppSet application junos-telnet
set applications application-set SrvAccessAppSet application cust-app
set security idp idp-policy ABC rulebase-ips rule ABC match application SrvAccessAppSet
set security idp idp-policy ABC rulebase-ips rule ABC then action no-action
set security idp active-policy ABC
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To create an application set and associate it with an IDP policy:

1. Create an application set and include three applications in the set.

   ```
   [edit applications application-set SrvAccessAppSet]
   user@host# set application junos-ssh
   user@host# set application junos-telnet
   user@host# set application cust-app
   ```

2. Create an IDP policy.

   ```
   [edit]
   user@host# edit security idp idp-policy ABC
   ```

3. Associate the application set with an IDP policy.

```
[edit security idp idp-policy ABC]
user@host# set rulebase-ips rule ABC match application SrvAccessAppSet
```

4. Specify an action for the policy.

```
[edit security idp idp-policy ABC]
user@host# set rulebase-ips rule ABC then action no-action
```

5. Activate the policy.

```
[edit]
user@host# set security idp active-policy ABC
```

Results From configuration mode, confirm your configuration by entering the **show security idp** and **show applications** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy ABC {
   rulebase-ips {
      rule R1 {
         match {
            application SrvAccessAppSet;
         }
         then {
            action {
               no-action;
            }
         }
      }
   }
   }
active-policy ABC;
```

```
[edit]
user@host# show applications
application-set SrvAccessAppSet {
   application ssh;
   application telnet;
   application custApp;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 220

Verifying the Configuration

Purpose Verify that the application set was associated with the IDP policy.

Action From operational mode, enter the **show security idp status** command.

See Also

- Understanding IDP Application Sets on page 218

Example: Configuring IDP Applications and Services

This example shows how to create an application and associate it with an IDP policy.

Requirements

Before you begin:

- Configure network interfaces.
- Enable IDP application services in a security policy. See "Example: Enabling IDP in a Security Policy" on page 45.

Overview

To create custom applications, specify a meaningful name for an application and associate parameters with it—for example, inactivity timeout, or application protocol type. In this example, you create a special FTP application called cust-app, specify it as a match condition in the IDP policy ABC running on port 78, and specify the inactivity timeout value as 6000 seconds.

Configuration

CLI Quick Configuration To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set applications application cust-app application-protocol ftp protocol tcp
   destination-port 78 inactivity-timeout 6000
set security idp idp-policy ABC rulebase-ips rule ABC match application cust-app
set security idp idp-policy ABC rulebase-ips rule ABC then action no-action
set security idp active-policy ABC
```

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To create an application and associate it with an IDP policy:

1. Create an application and specify its properties.

```
[edit applications application cust-app]
user@host# set application-protocol ftp protocol tcp destination-port 78
  inactivity-timeout 6000
```

2. Specify the application as a match condition in a policy.

```
[edit security idp idp-policy ABC rulebase-ips rule ABC]
user@host# set match application cust-app
```

3. Specify the no action condition.

```
[edit security idp idp-policy ABC rulebase-ips rule ABC]
user@host# set then action no-action
```

4. Activate the policy.

```
[edit]
user@host# set security idp active-policy ABC
```

Results From configuration mode, confirm your configuration by entering the **show security idp** and **show applications** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy ABC {
  rulebase-ips {
    rule R1 {
      match {
        application cust-app;
      }
    }
  }
  }
active-policy ABC;
```

```
[edit]
user@host# show applications
application cust-app {
  application-protocol ftp;
  protocol tcp;
  destination-port 78;
  inactivity-timeout 6000;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Configuration on page 223

Verifying the Configuration

Purpose Verify that the application was associated with the IDP policy.

Action From operational mode, enter the **show security idp status** command.

See Also

- Understanding IDP Application Sets on page 218

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

CHAPTER 4

Configuring IDP Features

IDP Application Identification

Supported Platforms SRX Series, vSRX

Juniper Networks provides predefined application signatures that detect Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) applications running on nonstandard ports.

For more information, see the following topics:

Understanding IDP Application Identification

Juniper Networks provides predefined application signatures that detect Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) applications running on nonstandard ports. Identifying these applications allows Intrusion Detection and Prevention (IDP) to apply appropriate attack objects to applications running on nonstandard ports. It also improves performance by narrowing the scope of attack signatures for applications without decoders.

The IDP sensor monitors the network and detects suspicious and anomalous network traffic based on specific rules defined in IDP rulebases. It applies attack objects to traffic based on protocols or applications. Application signatures enable the sensor to identify known and unknown applications running on nonstandard ports and to apply the correct attack objects.

Application signatures are available as part of the security package provided by Juniper Networks. You download predefined application signatures along with the security package updates. You cannot create application signatures. For information on downloading the security package, see “Updating the IDP Signature Database Manually Overview” on page 34.

On all branch SRX Series devices, the maximum supported number of entries in the ASC table is 100,000 entries. Because the user land buffer has a fixed size of 1 MB as a limitation, the table displays a maximum of 38,837 cache entries.

The maximum number of IDP sessions supported is 16,384 on SRX320 devices and 32,768 on SRX345 devices.

Application identification is enabled by default only if the service requesting the application identification (such as IDP, AppFW, AppTrack or AppQoS) is enabled to invoke the application identification. If none of these policies or configurations exist, application identification will not be automatically triggered. However, when you specify an application in the policy rule, IDP uses the specified application rather the application identification result. For instructions on specifying applications in policy rules, see “Example: Configuring IDP Applications and Services” on page 221.

NOTE: Application identification is enabled by default. To disable application identification with the CLI see *Disabling and Reenabling Junos OS Application Identification*.

On all branch SRX Series devices, IDP does not allow header checks for nonpacket contexts.

IDP deployed in both active/active and active/passive chassis clusters has the following limitations:

- No inspection of sessions that fail over or fail back.
- The IP action table is not synchronized across nodes.
- The Routing Engine on the secondary node might not be able to reach networks that are reachable only through a Packet Forwarding Engine.
- The SSL session ID cache is not synchronized across nodes. If an SSL session reuses a session ID and it happens to be processed on a node other than the one on which the session ID is cached, the SSL session cannot be decrypted and will be bypassed for IDP inspection.

IDP deployed in active/active chassis clusters has a limitation that for time-binding scope source traffic, if attacks from a source (with more than one destination) have active sessions distributed across nodes, then the attack might not be detected because time-binding counting has a local-node-only view. Detecting this sort of attack requires an RTO synchronization of the time-binding state that is not currently supported.

See Also

- Example: Configuring IDP Policies for Application Identification on page 229

Understanding IDP Service and Application Bindings by Attack Objects

Attack objects can bind to applications and services in different ways:

- Attack objects can bind to an application implicitly and not have a service definition. They bind to an application based on the name of a context or anomaly.
- Attack objects can bind to a service using a service name.
- Attack objects can bind to a service using TCP or UDP ports, ICMP types or codes or RPC program numbers.

Whether the specified application or service binding applies or not depends on the complete attack object definition as well as the IDP policy configuration:

- If you specify an application in an attack object definition, the service field is ignored. The attack object binds to the application instead of the specified service. However, if you specify a service and no application in the attack object definition, the attack object binds to the service. Table 87 on page 227 summarizes the behavior of application and service bindings with application identification.

Table 87: Applications and Services with Application Identification

Attack Object Fields	Binding Behavior	Application Identification
:application (http) **:service (smtp)**	• Binds to the application **HTTP**. • The service field is ignored.	Enabled
:service (http)	Binds to the application **HTTP**.	Enabled
:service (tcp/80)	Binds to TCP port 80.	Disabled

For example, in the following attack object definition, the attack object binds to the application **HTTP**, the application identification is enabled, and the service field **SMTP** is ignored.

```
: (“http-test”
   :application (“http”)
   :service (“smtp”)
   :rectype (signature)
   :signature (
     :pattern (“.*TERM=xterm; export TERM=xterm; exec bash – i\x0a\x.*”)
     :type (stream)
   )
   :type (attack-ip)
)
```

- If an attack object is based on service specific contexts (for example, **http-url**) and anomalies (for example, **tftp_file_name_too_long**), both application and service fields are ignored. Service contexts and anomalies imply application; thus when you specify these in the attack object, application identification is applied.

- If you configure a specific application in a policy, you overwrite the application binding specified in an attack object. Table 88 on page 228 summarizes the binding with the application configuration in the IDP policy.

Table 88: Application Configuration in an IDP Policy

Application Type in the Policy	Binding Behavior	Application Identification
Default	Binds to the application or service configured in the attack object definition.	• Enabled for application-based attack objects • Disabled for service-based attack objects
Specific application	Binds to the application specified in the attack object definition.	Disabled
Any	Binds to all applications.	Disabled

- If you specify an application in an IDP policy, the application type configured in the attack object definition and in the IDP policy must match. The policy rule cannot specify two different applications (one in the attack object and the other in the policy).

NOTE: Application cannot be any when attacks based on different applications are specified in IDP configuration and commit fails. Use default instead.

While configuring IDS rules for application the option any is deprecated.

But, when application is any and custom-attack groups are used in IDP configuration, commit goes through successfully. So, commit check does not detect such cases.

See Also

- Understanding IDP Application Identification on page 225
- Understanding the IDP Signature Database on page 31
- Example: Configuring IDP Policies for Application Identification on page 229

Understanding IDP Application Identification for Nested Applications

With the greater use of application protocol encapsulation, the need arises to support the identification of multiple different applications running on the same Layer 7 protocols. For example, applications such as Facebook and Yahoo Messenger can both run over HTTP, but there is a need to identify them as two different applications running on the same Layer 7 protocol. In order to do this, the current application identification layer is split into two layers: Layer 7 applications and Layer 7 protocols.

Included predefined application signatures have been created to detect the Layer 7 applications whereas the existing Layer 7 protocol signatures still function in the same manner. These predefined application signatures can be used in attack objects.

See Also

- Understanding IDP Application Identification on page 225

Example: Configuring IDP Policies for Application Identification

This example shows how to configure the IDP policies for application identification.

- Requirements on page 229
- Overview on page 229
- Configuration on page 229
- Verification on page 230

Requirements

Before you begin:

- Configure network interfaces.
- Download the application package.

Overview

In this example, you create an IDP policy ABC and define rule 123 in the IPS rulebase. You specify default as the application type in an IDP policy rule. If you specify an application instead of default the application identification feature will be disabled for this rule and IDP will match the traffic with the specified application type. The applications defined under application-identification cannot be referenced directly at this time.

Configuration

Step-by-Step Procedure

To configure IDP policies for application identification:

1. Create an IDP policy.

   ```
   [edit]
   user@host# set security idp idp-policy ABC
   ```

2. Specify the application type.

   ```
   [edit]
   user@host# set security idp idp-policy ABC rulebase-ips rule 123 match application
     default
   ```

3. Specify an action to take when the match condition is meet.

   ```
   [edit]
   user@host# set security idp idp-policy ABC rulebase-ips rule 123 then action
     no-action
   ```

4. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

Verification

To verify the configuration is working properly, enter the **show security idp** command.

See Also

- Understanding IDP Application Identification on page 225
- *Understanding the Junos OS Application Package Installation*

Understanding Memory Limit Settings for IDP Application Identification

Although you cannot create application signatures with the IDP signature database, you can configure sensor settings to limit the number of sessions running application identification and also limit memory usage for application identification.

Memory limit for a session—You can configure the maximum amount of memory bytes that can be used to save packets for application identification for one TCP or UDP session. You can also configure a limit for global memory usage for application identification. Application identification is disabled for a session after the system reaches the specified memory limit for the session. However, IDP continues to match patterns. The matched application is saved to cache so that the next session can use it. This protects the system from attackers trying to bypass application identification by purposefully sending large client-to-server packets.

- Number of sessions—You can configure the maximum number of sessions that can run application identification at the same time. Application identification is disabled after the system reaches the specified number of sessions. You limit the number of sessions so that you can prevent a denial-of-service (DOS) attack, which occurs when too many connection requests overwhelm and exhaust all the allocated resources on the system.

Table 89 on page 230 provides the capacity of a central point (CP) session numbers for SRX3400, SRX3600, SRX5600, and SRX5800 devices.

Table 89: Maximum CP Session Numbers

SRX Series Devices	Maximum Sessions	Central Point (CP)
SRX3400	2.25 million	Combo-mode CP
SRX3600	2.25 million	Combo-mode CP
SRX5600	9 million 2.25 million	Full CP Combo-mode CP

Table 89: Maximum CP Session Numbers (continued)

SRX Series Devices	Maximum Sessions	Central Point (CP)
SRX5800	10 million	Full CP
	2.25 million	Combo-mode CP

Example: Setting Memory Limits for IDP Application Identification Services

This example shows how to configure memory limits for IDP application identification services.

- Requirements on page 231
- Overview on page 231
- Configuration on page 231
- Verification on page 231

Requirements

Before you begin:

- Configure network interfaces.
- Download the signature database. See "Example: Updating the IDP Signature Database Manually" on page 35.

Overview

In this example, you configure 5000 memory bytes as the maximum amount of memory that can be used for saving packets for application identification for one TCP session.

Configuration

Step-by-Step Procedure

To configure memory and session limits for IDP application identification services:

1. Specify the memory limits for application identification.

   ```
   [edit]
   user@host# set security idp sensor-configuration application-identification
     max-tcp-session-packet-memory 5000
   ```

2. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

Verification

To verify the configuration is working properly, enter the **show security idp memory** command.

Verifying IDP Counters for Application Identification Processes

Purpose Verify the IDP counters for the application identification processes.

Action From the CLI, enter the **show security idp counters application-identification** command.

Sample Output

```
user@host> show security idp counters application-identification
IDP counters:

 IDP counter type                                                Value
 AI cache hits                                                   2682
 AI cache misses                                                 3804
 AI matches                                                      74
 AI no-matches                                                   27
 AI-enabled sessions                                             3804
 AI-disabled sessions                                            2834
 AI-disabled sessions due to cache hit                           2682
 AI-disabled sessions due to configuration                       0
 AI-disabled sessions due to protocol remapping                  0
 AI-disabled sessions due to non-TCP/UDP flows                   118
 AI-disabled sessions due to no AI signatures                    0
 AI-disabled sessions due to session limit                       0
 AI-disabled sessions due to session packet memory limit         34
 AI-disabled sessions due to global packet memory limit          0
```

Meaning The output shows a summary of the application identification counters. Verify the following information:

- AI cache hits—Displays the number of hits on the application identification cache
- AI cache misses—Displays the number of times the application matches but the application identification cache entry is not added.
- AI matches—Displays the number of times the application matches, and an application identification cache entry is added.
- AI no-matches—Displays the number of times when application does not match.
- AI-enabled sessions—Displays the number of sessions on which application identification is enabled.
- AI-disabled sessions—Displays the number of sessions on which application identification is disabled.
- AI-disabled sessions due to cache hit—Displays the number of sessions on which application identification is disabled after a cache entry is matched. Application identification process is discontinued for this session.
- AI-disabled sessions due to configuration—Displays the number of sessions on which application identification is disabled because of the sensor configuration.

- AI-disabled sessions due to protocol remapping—Displays the number of sessions for which application identification is disabled because you have configured a specific service in the IDP policy rule definition.
- AI-disabled sessions due to non-TCP/UDP flows—Displays the number of sessions for which application identification is disabled because the session is not a TCP or UDP session.
- AI-disabled sessions due to no AI signatures—Displays the number of sessions for which application identification is disabled because no match is found on the application identification signatures.
- AI-disabled due to session limit—Displays the number of sessions for which application identification is disabled because sessions have reached the maximum limit configured. Application identification is disabled for future sessions too.
- AI-disabled due to session packet memory limit—Displays the sessions for which application identification is disabled because sessions have reached the maximum memory limit on TCP or UDP flows. Application identification is disabled for future sessions too.
- AI-disabled due to global packet memory limit—Displays the sessions for which application identification is disabled because the maximum memory limit is reached. Application identification is disabled for future sessions too.

See Also

- Understanding IDP Application Identification on page 225

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

Class of Service Action in an IDP Policy

Supported Platforms SRX Series, vSRX

Class of Service (CoS) or Quality of Service (QoS) is a way to manage multiple traffic profiles over a network by giving certain types of traffic priority over others. For example you can give Voice traffic priority over email or http traffic.

For more information on IDP for CoS, see the following topics:

- IDP Class of Service Action Overview on page 234
- Forwarding Classes Overview on page 235
- Rewrite Rules Overview on page 237
- Example: Configuring and Applying Rewrite Rules on a Security Device on page 238
- Example: Applying the CoS Action in an IDP Policy on page 241

IDP Class of Service Action Overview

Differentiated Services (DS) is a system for tagging (or “marking”) traffic at a position within a hierarchy of priority. Differentiated Services codepoint (DSCP) marking maps the Junos OS Class of Service (CoS) level to the DSCP field in the IP packet header. On SRX1500 , SRX3400, SRX3600, SRX5600, and SRX5800 devices, DSCP values of IP packets can be rewritten by the following two software modules:

- Differentiated Services code point (DSCP) rewriter at an egress interface.
- IDP module according to IDP policies.

In the data plane, before a packet reaches an egress interface, the IDP module can notify the security flow module to rewrite the packet’s DSCP value. The IDP module and the interface-based rewriter rewrite DSCP values based on different and independent rules. The IDP module rewrites a packet’s DSCP value based on IDP policies; whereas the interface-based writer rewrites a packet’s DSCP value based on packet classification results. Therefore the rewriting decisions of the IDP module and the interface-based rewriter can be different.

An interface-based rewriter rewrites DSCP values by comparing a packet’s forwarding class against a set of forwarding classes configured as rewrite rules. A forwarding class that does not belong to this set of forwarding classes is used to notify an interface-based rewriter to not rewrite a packet’s DSCP value when it has been set by the IDP module.

NOTE: In addition to influencing the rewriting of a packet’s DSCP value, forwarding classes are also used to prioritize the traffic in the device. By assigning a forwarding class to a queue number, you affect the scheduling and marking of a packet as it transits an SRX Series device. For information on forwarding classes, see “Forwarding Classes Overview” on page 235.

When the IDP module rewrites a packet’s DSCP value, IDP can set the forwarding class associated with the packet such that the forwarding class is out of the set of forwarding classes defined as the rule for an egress interface-based rewriter. For information on rewrite rules, see “Rewrite Rules Overview” on page 237 and “Example: Configuring and Applying Rewrite Rules on a Security Device” on page 238.

When the interface-based rewriter processes the packet, it notices that the packet’s forwarding class does not match any of the classes defined in the rewrite rule, therefore it does not change the DSCP value of the packet. Consequently, the packet’s DSCP value is marked by the IDP module and the interface-based rewriter is bypassed. Separate forwarding classes for the IDP module and the interface-based rewriter can be defined using the **set forwarding-class** statement at the [edit class-of-service] hierarchy level. For example, forwarding classes fc0, fc1, fc2, and fc3 can be defined for the IDP module, while forwarding classes fc4, fc5, fc6, and fc7 can be defined for the interface-based rewriters. In Junos OS, multiple forwarding classes can be mapped to one priority queue. Therefore the number of forwarding classes can be more than the number of queues.

NOTE: When both the interface-based rewriter and the IDP modules try to rewrite DSCP values, the IDP module is given precedence over the interface-based rewriter because IDP marks DSCP values with more information about the packets and has stricter security criteria than the interface-based rewriter module.

For a configuration example that shows how you can rewrite DSCP values with the IDP module and bypass the interface-based rewriter, see "Example: Applying the CoS Action in an IDP Policy" on page 241.

See Also

- Example: Applying the CoS Action in an IDP Policy on page 241

Forwarding Classes Overview

Forwarding classes (FCs) allow you to group packets for transmission and to assign packets to output queues. The forwarding class and the loss priority define the per-hop behavior (PHB in DiffServ) of a packet.

Juniper Networks devices support eight queues (0 through 7). For a classifier to assign an output queue (default queues 0 through 3) to each packet, it must associate the packet with one of the following forwarding classes:

- Expedited forwarding (EF)—Provides a low-loss, low-latency, low-jitter, assured-bandwidth, end-to-end service.
- Assured forwarding (AF)—Provides a group of values you can define and includes four subclasses—AF1, AF2, AF3, and AF4—each with three drop probabilities (low, medium, and high).
- Best effort (BE)—Provides no service profile. For the BE forwarding class, loss priority is typically not carried in a class-of-service (CoS) value, and random early detection (RED) drop profiles are more aggressive.
- Network Control (NC)—This class is typically high priority because it supports protocol control.

In addition to behavior aggregate (BA) and mulitfield (MF) classification, the forwarding class (FC) of a packet can be directly determined by the logical interface that receives the packet. The packet FC can be configured using CLI commands, and if configured, this FC overrides the FC from any BA classification that was previously configured on the logical interface.

The following CLI command can assign an FC directly to packets received at a logical interface:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

This section contains the following topics:

- Forwarding Class Queue Assignments on page 236
- Forwarding Policy Options on page 237

Forwarding Class Queue Assignments

Juniper Networks devices have eight queues built into the hardware. By default, four queues are assigned to four FCs. Table 90 on page 236 shows the four default FCs and queues that Juniper Networks classifiers assign to packets, based on the class-of-service (CoS) values in the arriving packet headers.

NOTE: Queues 4 through 7 have no default assignments to FCs and are not mapped. To use queues 4 through 7, you must create custom FC names and map them to the queues.

By default, all incoming packets, except the IP control packets, are assigned to the FC associated with queue 0. All IP control packets are assigned to the FC associated with queue 3.

Table 90: Default Forwarding Class Queue Assignments

Forwarding Queue	Forwarding Class	Forwarding Class Description
Queue 0	best-effort (BE)	The Juniper Networks device does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. These packets are usually dropped under congested network conditions.
Queue 1	expedited-forwarding (EF)	The Juniper Networks device delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Devices accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
Queue 2	assured-forwarding (AF)	The Juniper Networks device offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile that you define. The device accepts excess traffic, but applies a random early detection (RED) drop profile to determine whether the excess packets are dropped and not forwarded. Three drop probabilities (low, medium, and high) are defined for this service class.

Table 90: Default Forwarding Class Queue Assignments (continued)

Forwarding Queue	Forwarding Class	Forwarding Class Description
Queue 3	network-control (NC)	The Juniper Networks device delivers packets in this service class with a low priority. (These packets are not delay sensitive.) Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, delay is preferable to discard.

Forwarding Policy Options

CoS-based forwarding (CBF) enables you to control next-hop selection based on a packet’s CoS and, in particular, the value of the IP packet's precedence bits. For example, you can specify a particular interface or next hop to carry high-priority traffic while all best-effort traffic takes some other path. CBF allows path selection based on FC. When a routing protocol discovers equal-cost paths, it can either pick a path at random or load-balance the packets across the paths, through either hash selection or round-robin selection.

A forwarding policy also allows you to create CoS classification overrides. You can override the incoming CoS classification and assign the packets to an FC based on the packets’ input interfaces, input precedence bits, or destination addresses. When you override the classification of incoming packets, any mappings you configured for associated precedence bits or incoming interfaces to output transmission queues are ignored.

See Also

- *Example: Assigning Forwarding Classes to Output Queues*
- *Example: Assigning a Forwarding Class to an Interface*
- *Example: Configuring Forwarding Classes*

Rewrite Rules Overview

A rewrite rule modifies the appropriate class-of-service (CoS) bits in an outgoing packet. Modification of CoS bits allows the next downstream device to classify the packet into the appropriate service group. Rewriting or marking outbound packets is useful when the device is at the border of a network and must alter the CoS values to meet the policies of the targeted peer. A rewrite rule examines the forwarding class and loss priority of a packet and sets its bits to a corresponding value specified in the rule.

Typically, a device rewrites CoS values in outgoing packets on the outbound interfaces of an edge device, to meet the policies of the targeted peer. After reading the current forwarding class and loss priority information associated with the packet, the transmitting device locates the chosen CoS value from a table, and writes this CoS value into the packet header.

NOTE: You can configure up to 32 IEEE 802.1p rewrite rules on each SRX5K-MPC on the SRX5600 and SRX5800 devices.

Example: Configuring and Applying Rewrite Rules on a Security Device

This example shows how to configure and apply rewrite rules for a device.

- Requirements on page 238
- Overview on page 238
- Configuration on page 239
- Verification on page 241

Requirements

Before you begin, create and configure the forwarding classes.

Overview

You can configure rewrite rules to replace CoS values on packets received from the customer or host with the values expected by other devices. You do not have to configure rewrite rules if the received packets already contain valid CoS values. Rewrite rules apply the forwarding class information and packet loss priority used internally by the device to establish the CoS value on outbound packets. After you configure rewrite rules, you must apply them to the correct interfaces.

In this example, you configure the rewrite rule for DiffServ CoS as rewrite-dscps. You specify the best-effort forwarding class as be-class, expedited forwarding class as ef-class, an assured forwarding class as af-class, and a network control class as nc-class. Finally, you apply the rewrite rule to an IRB interface.

NOTE: You can apply one rewrite rule to each logical interface.

Table 91 on page 238 shows how the rewrite rules replace the DSCPs on packets in the four forwarding classes.

Table 91: Sample rewrite-dscps Rewrite Rules to Replace DSCPs

mf-classifier Forwarding Class	For CoS Traffic Type	rewrite-dscps Rewrite Rules
be-class	Best-effort traffic—Provides no special CoS handling of packets. Typically, RED drop profile is aggressive and no loss priority is defined.	Low-priority code point: 000000 High-priority code point: 000001
ef-class	Expedited forwarding traffic—Provides low loss, low delay, low jitter, assured bandwidth, and end-to-end service. Packets can be forwarded out of sequence or dropped.	Low-priority code point: 101110 High-priority code point: 101111

Table 91: Sample rewrite-dscps Rewrite Rules to Replace DSCPs (continued)

mf-classifier Forwarding Class	For CoS Traffic Type	rewrite-dscps Rewrite Rules
af-class	Assured forwarding traffic—Provides high assurance for packets within the specified service profile. Excess packets are dropped.	Low-priority code point: 001010 High-priority code point: 001100
nc-class	Network control traffic—Packets can be delayed, but not dropped.	Low-priority code point: 110000 High-priority code point: 110001

NOTE: Forwarding classes can be configured in a DSCP rewriter and also as an action of an IDP policy to rewrite DSCP code points. To ensure that the forwarding class is used as an action in an IDP policy, it is important that you do not configure an IDP policy and interface-based rewrite rules with the same forwarding class.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class be-class
  loss-priority low code-point 000000
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class be-class
  loss-priority high code-point 000001
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class ef-class loss-priority
  low code-point 101110
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class ef-class loss-priority
  high code-point 101111
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class af-class loss-priority
  low code-point 001010
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class af-class loss-priority
  high code-point 001100
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class nc-class loss-priority
  low code-point 110000
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class nc-class loss-priority
  high code-point 110001
set class-of-service interfaces irb unit 0 rewrite-rules dscp rewrite-dscps
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure and apply rewrite rules for a device:

1. Configure rewrite rules for DiffServ CoS.

```
[edit]
user@host# edit class-of-service
user@host# edit rewrite-rules dscp rewrite-dscps
```

2. Configure best-effort forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class be-class loss-priority low code-point 000000
user@host# set forwarding-class be-class loss-priority high code-point 000001
```

3. Configure expedited forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class ef-class loss-priority low code-point 101110
user@host# set forwarding-class ef-class loss-priority high code-point 101111
```

4. Configure an assured forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class af-class loss-priority low code-point 001010
user@host# set forwarding-class af-class loss-priority high code-point 001100
```

5. Configure a network control class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class nc-class loss-priority low code-point 110000
user@host# set forwarding-class nc-class loss-priority high code-point 110001
```

6. Apply rewrite rules to an IRB interface.

```
[edit class-of-service]
user@host# set interfaces irb unit 0 rewrite-rules dscp rewrite-dscps
```

Results From configuration mode, confirm your configuration by entering the **show class-of-service** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show class-of-service
  interfaces {
  irb {
    unit 0 {
    rewrite-rules {
    dscp rewrite-dscps;
      }
    }
  }
  }
  rewrite-rules {
    dscp rewrite-dscps {
    forwarding-class be-class {
    loss-priority low code-point 000000;
      loss-priority high code-point 000001;
      }
      forwarding-class ef-class {
        loss-priority low code-point 101110;
      loss-priority high code-point 101111;
    }
    forwarding-class af-class {
      loss-priority low code-point 001010;
    loss-priority high code-point 001100;
  }
  forwarding-class nc-class {
    loss-priority low code-point 110000;
    loss-priority high code-point 110001;
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying Rewrite Rules Configuration

Purpose Verify that rewrite rules are configured properly.

Action From configuration mode, enter the **show class-of-service** command.

```
user@host> show class-of-service
Physical interface: irb, Index: 130
 Maximum usable queues: 8, Queues in use: 4
 Scheduler map: <default> , Index: 2
Congestion-notification: Disabled

Logical interface: irb.10, Index: 71
Object                  Name                   Type                    Index
Classifier              ipprec-compatibility   ip                        13
```

Meaning Rewrite rules are configured on IRB interface as expected.

See Also

- Rewrite Rules Overview on page 237

Example: Applying the CoS Action in an IDP Policy

As packets enter or exit a network, devices might be required to alter the CoS settings of the packet. Rewrite rules set the value of the CoS bits within the packet's header. In addition, you often need to rewrite a given marker (for example, DSCP) at the inbound interfaces of a device to accommodate BA classification by core devices.

On SRX Series devices, DSCP values of IP packets can be rewritten by the following two software modules:

- DSCP rewriter at an egress interface

- IDP module according to IDP policies

This example describes how to create an IDP policy that defines a forwarding class as an action item to rewrite the DSCP value of a packet.

- Requirements on page 242
- Overview on page 242
- Configuration on page 242
- Verification on page 247

Requirements

Before you begin, review the CoS components.

Overview

This example shows how you can rewrite DSCP values with the IDP module and bypass the interface-based rewriter. When you create an IDP policy to rewrite DSCP values, you must specify the following:

- Configure separate forwarding classes for the IDP module and the interface-based rewriters. In this example, eight forwarding classes, fc1 through fc8, are configured. Out of these eight forwarding classes, four classes, fc1 through fc4, are assigned to interface-based rewriters; the other four, fc5 through fc8, are assigned to the IDP module. These eight forwarding classes are mapped to four priority queues, queue 0 through queue 3.
- Configure the DSCP rewriter (rw_dscp) with forwarding classes, fc1 through fc4.
- Configure a DSCP classifier (c1) with the same forwarding classes as the DSCP rewriter. Essentially the classifier provides inputs, forwarding classes, and loss priorities to the rewriter.
- Apply the DSCP rewriter, rw_dscp, to a logical interface, ge-0/0/5.
- Apply the classifier, c1, to an ingress logical interface, ge-0/0/6.
- Create a new IDP policy (cos-policy) and assign class-of-service forwarding-class fc5 as the action.

NOTE: **To ensure DSCP rewriting by IDP, it is important that you do not configure an IDP policy and interface-based DSCP rewrite rules with the same forwarding class.**

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set class-of-service forwarding-classes queue 0 fc1
set class-of-service forwarding-classes queue 1 fc2
set class-of-service forwarding-classes queue 2 fc3
set class-of-service forwarding-classes queue 3 fc4
set class-of-service forwarding-classes queue 0 fc5
set class-of-service forwarding-classes queue 1 fc6
set class-of-service forwarding-classes queue 2 fc7
set class-of-service forwarding-classes queue 3 fc8
set class-of-service rewrite-rules dscp rw_dscp
set class-of-service rewrite-rules dscp rw_dscp forwarding-class fc1 loss-priority low
  code-point 000000
set class-of-service rewrite-rules dscp rw_dscp forwarding-class fc2 loss-priority low
  code-point 001000
set class-of-service rewrite-rules dscp rw_dscp forwarding-class fc3 loss-priority low
  code-point 010000
set class-of-service rewrite-rules dscp rw_dscp forwarding-class fc4 loss-priority low
  code-point 011000
set class-of-service classifiers dscp c1 forwarding-class fc1 loss-priority low code-points
  111111
set class-of-service classifiers dscp c1 forwarding-class fc2 loss-priority low code-points
  110000
set class-of-service classifiers dscp c1 forwarding-class fc3 loss-priority low code-points
  100000
set class-of-service classifiers dscp c1 forwarding-class fc4 loss-priority low code-points
  000000
set class-of-service interfaces ge-0/0/5 unit 0 rewrite-rules dscp rw_dscp
set class-of-service interfaces ge-0/0/6 unit 0 classifiers dscp c1
set security idp idp-policy cos-policy
set security idp idp-policy cos-policy rulebase-ips
set-security idp idp-policy cos-policy rulebase-ips rule r1
set-security idp idp-policy cos-policy rulebase-ips rule r1 match from-zone any to-zone
  any application default
set-security idp idp-policy cos-policy rulebase-ips rule r1 match attacks
  predefined-attack-groups 'P2P - All'
set security idp idp-policy cos-policy rulebase-ips rule r1 then action class-of-service
  forwarding-class fc5
set security idp idp-policy cos-policy rulebase-ips rule r1 then action class-of-service
  dscp-code-point 62
set security idp idp-policy cos-policy rulebase-ips rule r1 then notification log-attacks
set security idp idp-policy cos-policy rulebase-ips rule r1 then severity critical
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*in the *CLI User Guide*.

To configure an IDP policy that uses a forwarding class as a notification action for DSCP rewriting, perform the following tasks:

1. Configure forwarding classes.

   To configure a one-to-one mapping between the eight forwarding classes and the four priority queues, include the following statements at the **[edit class-of-service]** hierarchy level:

   ```
   [edit class-of-service]
   ```

```
user@host# set forwarding-classes fc1 queue-num 0
user@host# set forwarding-classes fc2 queue-num 1
user@host# set forwarding-classes fc3 queue-num 2
user@host# set forwarding-classes fc4 queue-num 3
user@host# set forwarding-classes fc5 queue-num 0
user@host# set forwarding-classes fc6 queue-num 1
user@host# set forwarding-classes fc7 queue-num 2
user@host# set forwarding-classes fc8 queue-num 3
```

2. Configure a DSCP rewriter with forwarding classes.

```
[edit class-of-service]
user@host# set rewrite-rules dscp rw_dscp forwarding-class fc1 loss-priority low
  code-point 000000
user@host# set rewrite-rules dscp rw_dscp forwarding-class fc2 loss-priority low
  code-point 001000
user@host# set rewrite-rules dscp rw_dscp forwarding-class fc3 loss-priority low
  code-point 010000
user@host# set rewrite-rules dscp rw_dscp forwarding-class fc4 loss-priority low
  code-point 011000
```

3. Configure a BA classifier with the same forwarding classes as the DSCP rewriter.

```
[edit class-of-service]
user@host# set classifiers dscp c1 forwarding-class fc1 loss-priority low code-points
  111111
user@host# set classifiers dscp c1 forwarding-class fc2 loss-priority low code-points
  110000
user@host# set classifiers dscp c1 forwarding-class fc3 loss-priority low code-points
  100000
user@host# set classifiers dscp c1 forwarding-class fc4 loss-priority low code-points
  000000
```

4. Apply the rewriter to a logical interface.

```
[edit class-of-service]
user@host# set interfaces ge-0/0/5 unit 0 rewrite-rules dscp rw_dscp
```

5. Apply the classifier to a logical interface.

```
[edit class-of-service]
user@host# set interfaces ge-0/0/6 unit 0 classifiers dscp c1
```

6. Configure the IDP policy with the action of forwarding class.

   The following steps show how an IDP policy includes a class-of-service forwarding class as one of the actions. In policy *cos-policy*, forwarding class fc5 is defined as an action in conjunction with the action of dscp-code-point 62, which requires the IDP module to rewrite DSCP values to 62. Taking actions of R1, the IDP module conducts the security flow module to rewrite the packets' DSCP values as 62 and set their forwarding classes as fc5.

To set a forwarding class as one of the actions in an IDP policy, perform the following tasks:

a. Create a policy by assigning a meaningful name to it.

```
[edit ]
user@host# edit security idp idp-policy cos-policy
```

b. Associate a rulebase with the policy.

```
[edit security idp idp-policy cos-policy ]
user@host# edit rulebase-ips
```

c. Add rules to the rulebase.

```
[edit security idp idp-policy cos-policy rulebase-ips]
user@host# edit rule R1
```

d. Define the match criteria for the rule.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set match from-zone any to-zone any application default
```

e. Define an attack as match criteria.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set match attacks predefined-attack-groups 'P2P - All'
```

f. Specify forwarding class as an action for the rule.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set then action class-of-service forwarding-class fc5
```

g. Specify dscp–code-point as an action for the rule.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set then action class-of-service dscp-code-point 62
```

h. Specify notification and logging options for the rule.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set then notification log-attacks alert
```

i. Set the severity level for the rule.

```
[edit security idp idp-policy cos-policy rulebase-ips rule R1]
user@host# set then severity critical
```

j. Activate the policy.

```
[edit]
user@host# set security idp active-policy cos-policy
```

Results From configuration mode, confirm your configuration by entering the **show security idp** and **show class-of-service** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
idp-policy cos-policy {
  rulebase-ips {
    rule R1 {
      match {
        from-zone any;
        to-zone any;
        application default;
        attacks {
          predefined-attack-groups P2P - All;
        }
      }
      then {
        action {
          class-of-service {
            forwarding-class fc5;
            dscp-code-point 62;
          }
        }
        notification {
          log-attacks {
            alert;
          }
        }
        severity critical;
      }
    }
  }
}
active-policy cos-policy;
```

```
[edit]
user@host# show class-of-service
classifiers {
  dscp c1 {
    forwarding-class fc1 {
      loss-priority low code-points 111111;
    }
    forwarding-class fc2 {
      loss-priority low code-points 110000;
    }
    forwarding-class fc3 {
      loss-priority low code-points 100000;
    }
    forwarding-class fc4 {
      loss-priority low code-points 000000;
    }
  }
}
  forwarding-classes {
    queue 0 fc5;
```

```
        queue 1 fc6;
        queue 2 fc7;
        queue 3 fc8;
    }
    interfaces {
        ge-0/0/5 {
            unit 0 {
                rewrite-rules {
                    dscp rw_dscp;
                }
            }
        }
        ge-0/0/6 {
            unit 0 {
                classifiers {
                    dscp c1;
                }
            }
        }
    }
    rewrite-rules {
        dscp rw_dscp {
            forwarding-class fc1 {
                loss-priority low code-point 000000;
            }
            forwarding-class fc2 {
                loss-priority low code-point 001000;
            }
            forwarding-class fc3 {
                loss-priority low code-point 010000;
            }
            forwarding-class fc4 {
                loss-priority low code-point 011000;
            }
        }
    }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform these tasks:

- Verifying IDP Policy Configuration on page 247
- Verifying CoS Configuration on page 248

Verifying IDP Policy Configuration

Purpose Verify that the forwarding class fc5 is configured as an action in the IDP policy.

Action From operational mode, enter the **show security idp idp-policy cos-policy** command.

Verifying CoS Configuration

Purpose Verify if the one-to-one mapping between the eight forwarding classes and the four priority queues, application of the BA classifier to the interfaces, and the rewrite rule are working.

Action From operational mode, enter the **show class-of-service** command.

See Also

- Understanding IDP Policy Rules on page 55
- Example: Enabling IDP in a Security Policy on page 45

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

IDP SSL Inspection

Supported Platforms SRX Series, vSRX

Secure Sockets Layer (SSL), also called Transport Layer Security (TLS), is a protocol suite for Web security that provides authentication, confidentiality and message integrity. Authentication guards against fraudulent transmissions by enabling a Web browser to validate the identity of a webserver. Confidentiality mechanisms ensure that communications are private. SSL enforces confidentiality by encrypting data to prevent unauthorized users from eavesdropping on electronic communications. Finally, message integrity ensures that the contents of a communication have not been tampered with.

For more information, see the following topics:

- IDP SSL Overview on page 248
- Supported IDP SSL Ciphers on page 249
- Understanding IDP Internet Key Exchange on page 250
- IDP Cryptographic Key Handling Overview on page 251
- Understanding IDP SSL Server Key Management and Policy Configuration on page 251
- Configuring an IDP SSL Inspection (CLI Procedure) on page 251
- Adding IDP SSL Keys and Associated Servers on page 252
- Deleting IDP SSL Keys and Associated Servers on page 252
- Displaying IDP SSL Keys and Associated Servers on page 253
- Example: Configuring IDP When SSL Proxy Is Enabled on page 253

IDP SSL Overview

Each SSL session begins with a handshake during which the client and server agree on the specific security key and the encryption algorithms to use for that session. At this time, the client also authenticates the server. Optionally, the server can authenticate the client. Once the handshake is complete, transfer of encrypted data can begin.

Juniper Networks provides Intrusion Detection and Prevention (IDP) SSL inspection that uses the SSL protocol suite consisting of different SSL versions, ciphers, and key exchange methods. Combined with the Application Identification feature, the SSL Inspection feature enables SRX Series devices to inspect HTTP traffic encrypted in SSL on any port. The following SSL protocols are supported:

- SSLv2
- SSLv3
- TLS

See Also

- IDP Policies Overview on page 43

Supported IDP SSL Ciphers

An SSL cipher comprises encryption cipher, authentication method, and compression. Junos OS supports all OPENSSL supported ciphers that do not involve the use of temporary private keys. For authentication, NULL, MD5, and SHA-1 authentication methods are supported.

NOTE: Compression and SSLv2 ciphers are not supported. Currently, most SSL servers automatically upgrade to a TLS cipher when an SSLv2 cipher is received in a client “hello” message. Check your browser to see how strong the ciphers can be and which ones your browser supports. (If the cipher is not in the list of supported ciphers, the session is ignored for deep packet inspection.)

Table 92 on page 249 shows the encryption algorithms supported by the SRX Series devices.

Table 92: Supported Encryption Algorithms

Cipher	Exportable	Type	Key Material	Expanded Key Material	Effective Key Bits	IV Size
NULL	No	Stream	0	0	0	N/A
DES-CBC-SHA	No	Block	8	8	56	8
DES-CBC3-SHA	No	Block	24	24	168	8
AES128-SHA	No	Block	16	16	128	16
AES256-SHA	No	Block	32	32	256	16

For more information on encryption algorithms, see *IPsec VPN Overview*.
Table 93 on page 250 shows the supported SSL ciphers.

Table 93: Supported SSL Ciphers

Cipher Suites	Value
TLS_RSA_WITH_NULL_MD5	0x0001
TLS_RSA_WITH_NULL_SHA	0x0002
TLS_RSA_WITH_DES_CBC_SHA	0x0009
TLS_RSA_WITH_3DES_EDE_CBC_SHA	0x000A
TLS_RSA_WITH_AES_128_CBC_SHA	0x002F
TLS_RSA_WITH_AES_256_CBC_SHA	0x0035

NOTE: RC4 and IDEA ciphers are not supported because of license and OPENSSL library availability.

Understanding IDP Internet Key Exchange

Internet Key Exchange (IKE) establishes a premaster secret that is used to generate symmetric keys for bulk data encryption and authentication. Section F.1.1 of RFC 2246 defines Transport Layer Security (TLS) authentication and key exchange methods. The two key exchange methods are:

- RSA—Rivest-Shamir-Adleman (RSA) is a key exchange algorithm that governs the way participants create symmetric keys or a secret that is used during an SSL session. The RSA key exchange algorithm is the most commonly used method.
- DSA—Digital Signature Algorithm (DSA) adds an additional authentication option to the IKE Phase 1 proposals. The DSA can be configured and behaves analogously to the RSA, requiring the user to import or create DSA certificates and configure an IKE proposal to use the DSA. Digital certificates are used for RSA signatures, DSA signatures, and the RSA public key encryption based method of authentication in the IKE protocol.
- Diffie-Hellman— Diffie-Hellman (DH) is a key exchange method that allows participants to produce a shared secret value. The strength of the technique is that it allows participants to create the secret value over an unsecured medium without passing the secret value through the wire.

The key exchange methods can use either a fixed or a temporary server key. IDP can successfully retrieve the premaster secret only if a fixed server key is used. For more information on Internet Key Exchange, see *Understanding Certificates and PKI*.

NOTE: Juniper IDP does not decrypt SSL sessions that use Diffie-Hellman key exchange.

IDP Cryptographic Key Handling Overview

With the Intrusion Detection and Prevention (IDP) Secure Sockets Layer (SSL) decryption feature, SRX Series devices load configured RSA private keys to memory and use them to establish SSL session keys to decrypt data. IDP is required to decrypt the RSA keys and to check the integrity before performing normal encryption or decryption operations using the keys.

The primary purpose of this feature is to ensure that RSA private keys used by IDP are not stored as plain text or in an easily understandable or usable format. The keys are decrypted to perform normal encryption or decryption operations. This feature also involves error detection checks during copying of the keys from one memory location to another, as well as overwriting of intermediate storage with nonzero patterns when the keys are no longer needed.

The **set security idp sensor-configuration ssl-inspection key-protection** CLI configuration command is used to enable this feature.

Understanding IDP SSL Server Key Management and Policy Configuration

The device can support up to 1000 server private keys. Each key can have up to 100 servers that use it. This capacity is the same regardless of the number of SPUs available on the device because essentially each SPU needs to be able to access all the keys.

Multiple servers can share the same private key; however, one server can have only one private key. SSL decryption is disabled by default. Both plain and encrypted keys are supported.

NOTE: Junos OS does not encrypt SSL keys file.

NOTE: You can set the value of SSL session ID cache timeout parameter by using the **set security idp sensor-configuration ssl-inspection session-id-cache-timeout** command. The default value of the cache timeout parameter is 600 seconds.

Configuring an IDP SSL Inspection (CLI Procedure)

SSL decoder is enabled by default. If you need to manually enable it via CLI, use the following CLI command.

```
set security idp sensor-configuration detector protocol-name SSL tunable-name sc_ssl_flags
  tuneable-value 1
```

To configure an IDP SSL inspection, use the following CLI procedure:

```
[edit security]
  idp {
  sensor-configuration {
    ssl-inspection {
            sessions <number>;
          }
        }
```

The sensor now inspects traffic for which it has a key/server pair.

NOTE: Maximum supported sessions per SPU: default value is 10,000 and range is 1 through 100,000. The session limit is per SPU, and it is the same regardless of the number of SPUs on the device.

Adding IDP SSL Keys and Associated Servers

When you are installing a key, you can password protect the key and also associate it to a server.

To install a Privacy-Enhanced Mail (PEM) key, use the following CLI command:

request security idp ssl-inspection key add *key-name* file *file-path* server *server-ip* password *password-string*

NOTE: In a two-node SRX Series cluster, the key has to be manually copied over to both Node 0 and Node 1 at the same location for the request command to be successful.

You can also associate the key with a server at a later time by using the add server CLI command. A server can be associated with only one key. To associate a server to the installed key, use the following CLI command:

request security idp ssl-inspection key add *key-name* server *server-ip*

NOTE: The maximum key name length is 32 bytes, including the ending “\0”.

Deleting IDP SSL Keys and Associated Servers

- To delete all keys and servers, use the following CLI command:

  user@host> **request security idp ssl-inspection key delete**

  All installed keys are deleted along with any associated servers.

- To delete a specific key and all associated servers with that key, use the following CLI command:

  user@host> **request security idp ssl-inspection key delete** <key-name>

  Deletes the specified key and all servers associated with that key.

- To delete a single server, use the following CLI command:

  user@host> **request security idp ssl-inspection key delete** <key-name> **server** <server-ip>

Deletes the specified server that is bound to the specified key.

Displaying IDP SSL Keys and Associated Servers

- To display all installed server keys and associated server, use the following CLI command:

  ```
  user@host> show security idp ssl-inspection key
  ```

  Displays all server keys and IP addresses bound to those keys. The following example shows CLI output when the **show security idp ssl-inspection key** command is used:

  ```
  Total SSL keys : 2
  SSL server key and ip address :
    Key : key1, server : 1.1.1.1
    Key : key2, server : 2.2.2.2
    Key : key2, server : 2.2.2.3
  ```
- To display IP addresses bound to a specific key, use the following CLI command:

  ```
  user@host> show security idp ssl-inspection key <key-name>
  ```

  The following is an example of the CLI output received when the **show security idp ssl-inspection key <key-name>** command is used:

  ```
  Key : key1, server : 1.1.1.1
  ```

Example: Configuring IDP When SSL Proxy Is Enabled

This example describes how IDP supports the application identification (AppID) functionality when SSL proxy is enabled.

- Requirements on page 253
- Overview on page 254
- Configuration on page 254
- Verification on page 254

Requirements

Before you begin:

- Create zones. See *Example: Creating Security Zones*.
- Configure an address book with addresses for the policy. See *Example: Configuring Address Books and Address Sets*.
- Create an application (or application set) that indicates that the policy applies to traffic of that type. See *Example: Configuring Applications and Application Sets*.
- Create an SSL proxy profile that enables SSL proxy by means of a policy. See *Configuring SSL Forward Proxy*.
- Configure an IDP policy as an active policy. See “Example: Enabling IDP in a Security Policy” on page 45

Overview

This example shows how to configure IDP in a policy rule when SSL proxy is enabled.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security policies from-zone Z_1 to-zone Z_2 policy policy1 match source-address any
set security policies from-zone Z_1 to-zone Z_2 policy policy1 match destination-address
  any
set security policies from-zone Z_1 to-zone Z_2 policy policy1 match application junos-https
set security policies from-zone Z_1 to-zone Z_2 policy policy1 then permit
  application-services ssl-proxy profile-name ssl-profile-1
set security policies from-zone Z_1 to-zone Z_2 policy policy1 then permit
  application-services idp
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

In this example, you configure a security policy that uses IDP as the application service.

1. Configure a policy to process the traffic with SSL proxy profile ssl-profile-1.

   ```
   [edit security policies from-zone Z_1 to-zone Z_2 policy policy1
   user@host# set match source-address any
   user@host# set match destination-address any
   user@host# set match application junos-https
   user@host# set then permit application-services ssl-proxy profile-name ssl-profile-1
   ```

2. Define IDP as the application service.

   ```
   [edit security policies from-zone Z_1 to-zone Z_2 policy policy1
   user@host#  set then permit application-services idp
   ```

Results

From configuration mode, confirm your configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Verification

Verify that the configuration is working properly. Verification in IDP is similar to verification in Application Firewall. See *Example: Configuring Application Firewall When SSL Proxy Is Enabled*.

See Also

- *SSL Proxy Overview*

- *Application Firewall, IDP, and Application Tracking with SSL Proxy Overview*
- *Understanding Security Policy Elements*
- *Security Policies Configuration Overview*

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

CHAPTER 5

Monitoring IDP

IDP Event Logging

Supported Platforms SRX Series, vSRX

The basic Junos OS system logging continues to function after Intrusion Detection and Prevention (IDP) is enabled.

For more information, see the following topics:

- Understanding IDP Logging on page 257
- Understanding IDP Log Suppression Attributes on page 258
- Example: Configuring IDP Log Suppression Attributes on page 259
- Understanding IDP Log Information Usage on the IC Series UAC Appliance on page 260
- IDP Alarms and Auditing on page 260

Understanding IDP Logging

An IDP-enabled device continues to record events that occur because of routine operations, such as a user login into the configuration database. It records failure and error conditions, such as failure to access a configuration file. You can configure files to log system messages and also assign attributes, such as severity levels, to messages. In addition to the regular system log messages, IDP generates event logs for attacks.

IDP generates event logs when an event matches an IDP policy rule in which logging is enabled. When you configure a rule for logging, the device creates a log entry for each event that matches that rule. You can use the CLI or J-Web to configure the policy rules to generate event logs.

NOTE: In the IDP attack detection event log message (IDP_ATTACK_LOG_EVENT_LS), the time-elapsed, inbytes, outbytes, inpackets, and outpackets fields are not populated.

Because IDP event logs are generated during an attack, log generation happens in bursts, generating a much larger volume of messages during an attack. In comparison to other event messages, the message size is also much larger for attack generated messages. The log volume and message size are important concerns for log management. To better manage the volume of log messages, IDP supports log suppression.

By configuring log suppression you can suppress multiple instances of the same log occurring from the same or similar sessions over the same period of time. Enabling log suppression ensures that minimal numbers of logs are generated for the same event or attack that occurs multiple times.

See Also

- IDP Policies Overview on page 43
- Understanding Security Packet Capture on page 273
- Understanding IDP Log Information Usage on the IC Series UAC Appliance on page 260

Understanding IDP Log Suppression Attributes

Log suppression ensures that minimal numbers of logs are generated for the same event or attack that occurs multiple times. Log suppression is enabled by default. You can configure certain log suppression attributes to suppress logs according to your needs. When configuring log suppression, keep in mind that log suppression can negatively impact sensor performance if you set the reporting interval too high.

You can configure the following log suppression attributes:

- Include destination addresses while performing log suppression—You can choose to combine log records for events with a matching source address. By default, the IDP sensor does not consider destination when matching events for log suppression.
- Number of log occurrences after which log suppression begins—You can specify the number of instances that a specific event must occur before log suppression begins. By default, log suppression begins after the first occurrence.
- Maximum number of logs that log suppression can operate on—When log suppression is enabled, Intrusion Detection and Prevention (IDP) must cache log records so that it can identify when multiple occurrences of the same event occur. You can specify how many log records are tracked simultaneously by IDP. By default, the maximum number of log records that IDP can operate on is 16,384.
- Time after which suppressed logs are reported—When log suppression is enabled, IDP maintains a count of occurrences of the same event. After the specified number of seconds have passed, IDP writes a single log entry containing the count of occurrences. By default, IDP reports suppressed logs after 5 seconds.

Example: Configuring IDP Log Suppression Attributes

This example shows how to configure log suppression attributes.

Requirements

Before you begin:

- Configure network interfaces.
- Download the signature database. See "Updating the IDP Signature Database Manually Overview" on page 34.

Overview

Log suppression ensures that minimal numbers of logs are generated for the same event or attack that occurs multiple times. Log suppression is enabled by default. You can configure certain log suppression attributes to suppress logs according to your needs.

In this example, you configure log suppression to begin after the second occurrence of an event and specify that logs are reported after 20 seconds.

Configuration

Step-by-Step Procedure

To configure log suppression attributes:

1. Specify the log number after which you want to start log suppression.

```
[edit]
user@host# set security idp sensor-configuration log suppression start-log 2
```

2. Specify the maximum time after which suppressed logs are reported.

```
[edit]
user@host# set security idp sensor-configuration log suppression max-time-report
  20
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

Verification

To verify log statistics, enter the **show security idp counters log** command.

See Also

- Updating the IDP Signature Database Manually Overview on page 34
- Example: Defining Rules for an IDP IPS RuleBase on page 66
- Understanding IDP Log Suppression Attributes on page 258

Understanding IDP Log Information Usage on the IC Series UAC Appliance

The IC Series UAC Appliance for the Unified Access Control (UAC) appliance can use Intrusion Detection and Prevention (IDP) attack log information sent from the Juniper Networks device to apply access policies for traffic in which IDP logs indicate an attack has been detected. Using a secure channel of communication, these IDP logs are sent to the IC Series appliance directly and securely. IDP attack logs are sent to the IC Series appliance through the JUEP communication channel.

This topic contains the following sections:

Message Filtering to the IC Series UAC Appliance

When you configure the IC Series UAC Appliance to receive IDP log messages, you set certain filtering parameters on the IC Series appliance. Without this filtering, the IC Series appliance could potentially receive too many log messages. The filtering parameters could include the following:

- The IC Series appliance should only receive communications from IDP for sessions it has authenticated. See the *Unified Access Control Administration Guide* for details.
- You can create IC Series appliance filters for receiving IDP logs files based on the their severity. For example, if on the IC Series appliance the severity is set to high, then IDP only sends logs which have a severity greater than or equal to high. See the *Unified Access Control Administration Guide* for details.
- From the IC Series appliance, you can disable the receiving of all IDP logs. See the *Unified Access Control Administration Guide* for details.

Configuring IC Series UAC Appliance Logging

All the configuration for receiving and filtering IDP logs is done on the IC Series UAC Appliance. You should refer to the *Unified Access Control Administration Guide* for configuration information for receiving IDP logs and details on the JUEP communication channel.

IDP Alarms and Auditing

By default, IDP logs the occurrence of an event without raising an alarm to the administrator. When the system is configured to log an event and the **potential-violation** option is set, IDP logs on the Packet Forwarding Engine are forwarded to Routing Engine. The Routing Engine then parses the IDP attack logs and raises IDP alarms as necessary.

- To enable an IDP alarm, use the **set security alarms potential-violation idp** command.
- To verify that the configuration is working properly, use the **show security alarms** command.

NOTE: In releases before Junos OS Release 11.2, IDP attack logs contain information about an attack event but do not raise alarms to the administrator.

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

IDP Sensor Configuration

Supported Platforms SRX Series, vSRX

Although you cannot create application signatures with the IDP signature database, you can configure sensor settings to limit the number of sessions running application identification and also to limit memory usage for application identification.

For more information, see the following topics:

- Understanding IDP Sensor Configuration Settings on page 261
- Example: Improving Logging and Traffic Analysis with IDP Sensor Configuration Options on page 267

Understanding IDP Sensor Configuration Settings

Sensor configuration options are used to:

- Log run conditions as IDP session capacity and memory limits are approached.
- To analyze traffic dropped by IDP and application identification when the limits are exceeded.

Although you cannot create application signatures with the IDP signature database, you can configure sensor settings to limit the number of sessions running application identification and also to limit memory usage for application identification.

You can configure the maximum amount of memory bytes that can be used to save packets for application identification for one TCP or UDP session. You can also configure a limit for global memory usage for application identification. Application identification is disabled for a session after the system reaches the specified memory limit for the session. However, IDP continues to match patterns. The matched application is saved to cache so that the next session can use it. This protects the system from attackers trying to bypass application identification by purposefully sending large client-to-server packets.

- **max-tcp-session-packet-memory**—To configure memory and session limits for IDP application identification services, run the **set security idp sensor-configuration application-identification max-tcp-session-packet-memory 5000** command.
- **memory-limit-percent**—To set memory limit percentage for data plane available in the system, which can be used for IDP allocation, run the **set security idp**

sensor-configuration global memory-limit-percent command. The supported percentage value is from 10 through 90.

- **drop-if-no-policy-loaded**—At startup, traffic is ignored by IDP by default if the IDP policy is not yet loaded. The **drop-if-no-policy-loaded** option changes this behavior so that all sessions are dropped before the IDP policy is loaded.

  The following counter for the **show security idp counters flow** command output analyzes dropped traffic due to the **drop-if-no-policy-loaded** option:

  ```
  Sessions dropped due to no policy                      0
  ```

- **drop-on-failover**—By default, IDP ignores failover sessions in an SRX Series chassis cluster deployment. The **drop-on-failover** option changes this behavior and automatically drops sessions that are in the process of being inspected on the primary node when a failover to the secondary node occurs.

  The following counter for the **show security idp counters flow** command output analyzes dropped failover traffic due to the **drop-on-failover** option:

  ```
  Fail-over sessions dropped                             0
  ```

- **drop-on-limit**—By default, sessions are not dropped if the IDP session limit or resource limits are exceeded. In this case, IDP and other sessions are dropped only when the device’s session capacity or resources are depleted. The **drop-on-limit** option changes this behavior and drops sessions when resource limits are exceeded.

  The following counters for the **show security idp counters flow** command output analyze dropped IDP traffic due to the **drop-on-limit** option:

  ```
  SM Sessions encountered memory failures                0
  SM Packets on sessions with memory failures            0
  SM Sessions dropped                                    0
  Both directions flows ignored                          0
  IDP Stream Sessions dropped due to memory failure      0
  IDP Stream Sessions ignored due to memory failure      0
  IDP Stream Sessions closed due to memory failure       0
  Number of times Sessions exceed high mark              0
  Number of times Sessions drop below low mark           0
  Memory of Sessions exceeds high mark                   0
  Memory of Sessions drops below low mark                0
  ```

  The following counters for the **show security idp counters application-identification** command output analyze dropped application identification traffic due to the **drop-on-limit** option:

  ```
  AI-session dropped due to malloc failure before session create        0
  AI-Sessions dropped due to malloc failure after create                0
  AI-Packets received on sessions marked for drop due to malloc failure 0
  ```

The following options are used to trigger informative log messages about current run conditions. When set, the log messages are triggered whether the **drop-on-limit** option is set or not.

- **max-sessions-offset**—The **max-sessions-offset** option sets an offset for the maximum IDP session limit. When the number of IDP sessions exceeds the maximum session limit, a warning is logged that conditions exist where IDP sessions could be dropped. When the number of IDP sessions drops below the maximum IDP session limit minus the offset value, a message is logged that conditions have returned to normal.

```
Jul 19 04:38:13 4.0.0.254 RT_IDP: IDP_SESSION_LOG_EVENT: IDP: at 1374233893,
FPC 4 PIC 1 IDP total sessions pass through high mark 100000. IDP may drop new
 sessions. Total sessions dropped 0.
```

```
Jul 19 04:38:21 4.0.0.254 RT_IDP: IDP_SESSION_LOG_EVENT: IDP: at 1374233901,
FPC 4 PIC 1 IDP total sessions drop below low mark 99000. IDP working in normal
 mode. Total sessions dropped 24373.
```

- **min-objcache-limit-lt**—The **min-objcache-limit-lt** option sets a lower threshold for available cache memory. The threshold value is expressed as a percentage of available IDP cache memory. If the available cache memory drops below the lower threshold level, a message is logged stating that conditions exist where IDP sessions could be dropped because of memory allocation failures. For example, the following message shows that the IDP cache memory has dropped below the lower threshold and that a number of sessions have been dropped:

```
Jul 19 04:07:33 4.0.0.254 RT_IDP: IDP_SESSION_LOG_EVENT: IDP: at 1374232053,
FPC 4 PIC 1 IDP total available objcache(used 4253368304, limit 7247757312)
drops below low mark 3986266515. IDP may drop new sessions. Total sessions
dropped 1002593.
```

- **min-objcache-limit-ut**—The **min-objcache-limit-ut** option sets an upper threshold for available cache memory. The threshold value is expressed as a percentage of available IDP cache memory. If available IDP cache memory returns to the upper threshold level, a message is logged stating that available cache memory has returned to normal. For example, the following message shows that the available IDP cache memory has increased above the upper threshold and that it is now performing normally:

```
Jul 19 04:13:47 4.0.0.254 RT_IDP: IDP_SESSION_LOG_EVENT: IDP: at 1374232428,
FPC 4 PIC 1 IDP total available objcache(used 2782950560, limit 7247757312)
increases above high mark 4348654380. IDP working in normal mode. Total sessions
 dropped 13424632.
```

NOTE: This message is triggered only if the lower threshold has been reached and the available memory has returned above the upper threshold. Fluctuations in available memory that dropped below the upper threshold but did not fall below the lower threshold do not trigger the message.

Starting with Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, IDP Intelligent Bypass feature is supported on SRX Series.

In its default configuration, IDP attempts to inspect new and existing sessions, regardless of CPU utilization. This can lead to dropped packets, latency, and instability across the system during high CPU utilization events. To overcome unpredictable IDP packet processing behavior, you can enable the IDP Intelligent Bypass feature. This feature will give you the flexibility to bypass IDP or to drop the packets when the system CPU utilization reaches a high level, otherwise known as “Failing Open” (permit packets) or “Failing Closed” (dropping packets). By default, IDP Intelligent Bypass feature is not enabled. The following options are used to configure the IDP Intelligent Bypass feature.

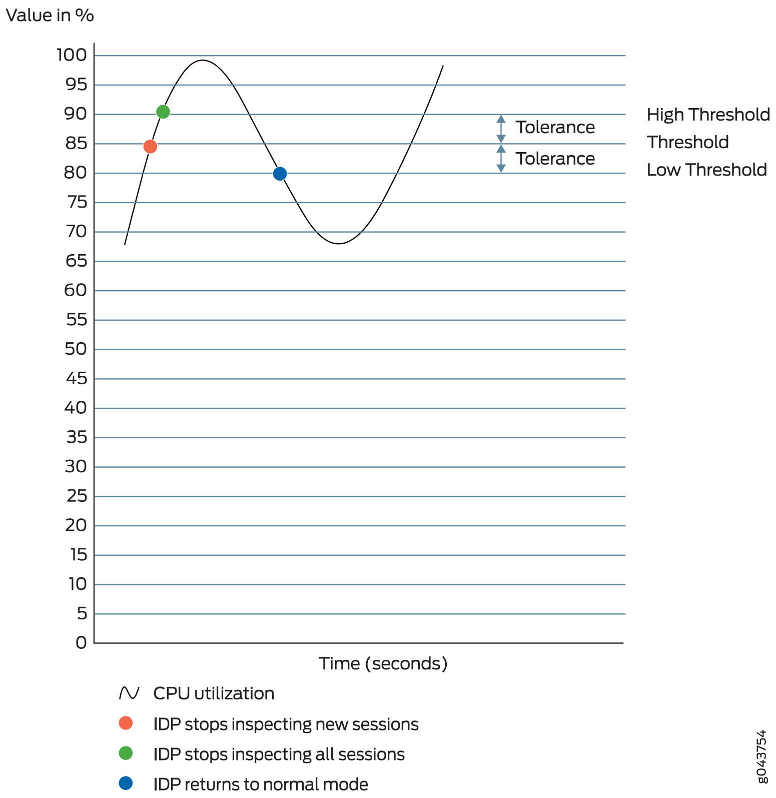
- **idp-bypass-cpu-usage-overload**— By default, IDP may consume 100 percent of available CPU and may begin dropping packets for all sessions inadvertently. To handle IDP packet processing behavior when the system CPU utilization reaches high threshold value, you can enable the IDP Intelligent Bypass feature. To enable IDP Intelligent Bypass feature, issue the **set security idp sensor-configuration flow idp-bypass-cpu-overload** command. By default, IDP Intelligent Bypass feature is not enabled.
- **idp-bypass-cpu-threshold**— IDP stops inspecting new sessions when CPU utilization reaches the defined threshold value. The default threshold CPU utilization value is 85 percent. When CPU utilization reaches threshold value, IDP keeps on bypassing new sessions until CPU utilization falls below the lower threshold value. Alternatively, if you set the **drop-on-limit**, where IDP drops new session until CPU utilization falls below the lower threshold value. To configure the threshold value, issue **set security idp sensor-configuration flow idp-bypass-cpu-threshold** command. You can set a threshold value in the range 0 through 99. This threshold value is expressed as a percentage.
- **idp-bypass-cpu-tolerance**— To configure the tolerance value, issue the **set security idp sensor-configuration flow idp-bypass-cpu-tolerance** command. You can set a tolerance value in the range 1 through 99. The default tolerance value is 5. This tolerance value is expressed as a percentage.

You can calculate the CPU upper and lower threshold values by using the following equations:

CPU upper threshold value = CPU threshold + CPU tolerance value.

CPU lower threshold value = CPU threshold - CPU tolerance value.

Figure 1: Understanding IDP Packet Processing Behavior During High Threshold



When the system CPU utilization exceeds the threshold value, IDP stops inspecting new sessions, but continues to inspect existing sessions. In this state, if **drop-on-limit** is set, IDP starts dropping new sessions. Log messages are triggered to indicate new sessions are dropped. For example, the following message states that IDP CPU utilization has crossed the threshold value and IDP may drop new sessions:

```
FPC 0 PIC 1 IDP CPU usage 86 crossed threshold value 85. IDP may drop new sessions.
 Total sessions dropped 2
```

When the system CPU utilization exceeds the upper threshold value, IDP stops inspecting the packets of existing sessions and new sessions. In this state, no packets can go through IDP inspection. If **drop-on-limit** is set, IDP drops all sessions. Log messages are triggered to indicate all sessions are dropped. For example, the following message states that IDP CPU utilization has crossed the upper threshold value, and IDP stops inspecting the packets of existing sessions and new sessions:

```
FPC 0 PIC 1 IDP CPU usage 92 crossed upper threshold value 90. IDP may drop packets
 of existing sessions as well as new sessions. Total sessions dropped 21
```

When the system CPU utilization falls below the lower threshold value, IDP starts inspecting new session and returns to normal mode. IDP will not inspect existing discarded sessions. Log messages are triggered to indicate IDP starts inspecting new session and returned to normal mode. For example, the following message states that IDP CPU utilization falls below the lower threshold value, and IDP returns to normal mode:

```
FPC 0 PIC 1 IDP CPU usage 75 dropped below lower threshold value 80. IDP working
 in normal mode. Total sessions dropped 25
```

IDP Protection Modes

IDP protection modes adjust the inspection parameters for efficient inspection of traffic in the device. To enable the IDP protection modes, issue the **security-configuration protection-mode *mode*** command at the **[edit security idp sensor-configuration]** hierarchy level.

user@host# set security-configuration protection-mode *mode*

There are four IDP protection modes :

NOTE: **All IDP protection modes inspect CTS(Client To Server) traffic.**

Table 94:

Mode	Description
Perimeter-Full	Inspects all STC(Server To Client) traffic. Processes TCP errors without any optimization. NOTE: This is the default mode.
Perimeter	Inspects all STC traffic. Processes TCP errors with optimization. For TCP packets, if SYN is received in a window and has a TCP error flag set, then process the TCP error and take appropriate action. Drop the current packet and ignore inspection on the entire session.
Datacenter-Full	Disables all STC traffic inspection. Processes TCP errors without any optimization. NOTE: Datacenter-Full can be used in situations where the SRX device is only responsible for protecting servers whose response traffic is not deemed interesting for analysis. Datacenter-Full should not be used in cases where the SRX device is responsible for protecting clients.
Datacenter	Disables all STC traffic inspection. Processes TCP errors with optimization. For TCP packets, if SYN is received in a window and has a TCP error flag set, then process the TCP error and take appropriate action. Drop the current packet and ignore inspection on the entire session. Datacenter configuration is optimized to provide balanced protection and performance.

See Also

- Understanding IDP Application Identification on page 225

Example: Improving Logging and Traffic Analysis with IDP Sensor Configuration Options

This example shows how to improve logging and traffic analysis by configuring IDP sensor configuration options. For instance, although you cannot create application signatures with the IDP signature database, you can configure sensor settings to limit the number of sessions running application identification and to limit its memory usage. In addition, you can use these options to log run conditions as IDP session capacity and memory limits are approached, and to analyze traffic dropped by IDP and application identification when exceeding these limitations.

Requirements

Before you begin:

- Configure network interfaces.
- Download the signature database. See "Example: Updating the IDP Signature Database Manually" on page 35. Application signatures are available as part of the security package provided by Juniper Networks. You download predefined application signatures along with the security package updates.

Overview

The IDP sensor monitors the network and detects suspicious and anomalous network traffic based on specific rules defined in IDP rulebases. It applies attack objects to traffic based on protocols or applications. Application signatures enable the sensor to identify known and unknown applications running on nonstandard ports and to apply the correct attack objects.

The default behavior of IDP is to ignore the sessions when:

- IDP policy is not configured in the device
- Resource limits (memory or active sessions) are reached
- In case of Chassis Cluster, for failed over sessions

If traffic availability is considered more important than security, then it is recommended to continue to use the above mentioned default behavior of IDP. However, If security is considered more important than availability, then it is recommended to change the default behavior with the configuration provided in this example.

You can achieve the following from this example:

- Although you cannot create application signatures with the IDP signature database, you can configure sensor settings to limit the number of sessions running application identification and also limit memory usage for application identification. You can configure the maximum amount of memory bytes that can be used to save packets for application identification for one TCP or UDP session. You can also configure a limit for global memory usage for application identification. Application identification is disabled for a session after the system reaches the specified memory limit for the session.
- By default, IDP ignores failover sessions that are in the process of being inspected on the primary node when a failover to the secondary node occurs in an SRX Series chassis cluster deployment. In this example, you specify that these sessions are dropped automatically and are captured in the respective counter instead of being ignored. You can monitor and analyze the sessions dropped when a failover on the secondary node occurs.
- By default, sessions are not dropped if the IDP session limit or resource limits are exceeded. In this example, you specify that if the IDP session limit or resource limits are exceeded, then the sessions are dropped and logging is added. You can set a maximum sessions offset limit value for the maximum IDP session limit. When the number of IDP sessions exceeds that value, a warning is logged that conditions exist where IDP sessions could be dropped. When the number of IDP sessions drops below the maximum IDP session limit minus the offset value, a message is logged that conditions have returned to normal.
- You can specify a lower threshold for available cache memory. If the available cache memory drops below the lower threshold level, a message is logged stating that conditions exist where IDP sessions could be dropped because of memory allocation failures. This log enables you to control the number of sessions dropped, and these dropped sessions can later be analyzed and considered for processing.
- Similarly, you can specify an upper threshold for available cache memory. If available IDP cache memory returns to the upper threshold level, a message is logged stating that available cache memory has returned to normal. This log enables you to control the number of sessions dropped, and these dropped sessions can later be analyzed and considered for processing.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp sensor-configuration application-identification
  max-tcp-session-packet-memory 5000
set security idp sensor-configuration flow drop-if-no-policy-loaded
set security idp sensor-configuration flow drop-on-failover
set security idp sensor-configuration flow drop-on-limit
set security idp sensor-configuration flow max-sessions-offset 5
set security idp sensor-configuration flow min-objcache-limit-lt 21
set security idp sensor-configuration flow min-objcache-limit-ut 56
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To set IDP sensor configuration options:

1. Specify the memory limits for application identification.

   ```
   [edit security idp sensor-configuration]
   user@host# set application-identification max-tcp-session-packet-memory 5000
   ```

2. Specify that traffic is dropped before the IDP policy is loaded.

   ```
   [edit security idp sensor-configuration flow]
   user@host# set drop-if-no-policy-loaded
   ```

3. Specify that failover sessions in an SRX Series chassis cluster deployment are dropped.

   ```
   [edit security idp sensor-configuration flow]
   user@host# set drop-on-failover
   ```

4. Specify that sessions are dropped when resource limits are exceeded.

   ```
   [edit security idp sensor-configuration flow]
   user@host# set drop-on-limit
   ```

   NOTE: If you do not want the sessions to be dropped when resource limits are exceeded, run the **delete drop-on-limit** command.

5. Configure an offset value for the maximum IDP session limit.

   ```
   [edit ssecurity idp sensor-configuration flow]
   user@host# set max-sessions-offset 5
   ```

6. Set a lower threshold for available cache memory.

   ```
   [edit security idp sensor-configuration flow]
   user@host# set min-objcache-limit-lt 21
   ```

7. Set an upper threshold for available cache memory.

   ```
   [edit security idp sensor-configuration flow]
   user@host# set min-objcache-limit-ut 56
   ```

Results

From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp
sensor-configuration {
  application-identification {
    max-tcp-session-packet-memory 5000;
  }
  flow {
    drop-on-limit;
    drop-on-failover;
    drop-if-no-policy-loaded;
    max-sessions-offset 5;
    min-objcache-limit-lt 21;
    min-objcache-limit-ut 56;
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying IDP Sensor Configuration Settings

Purpose Verify the IDP sensor configuration settings.

Action From operational mode, enter the **show security idp sensor-configuration** command.

```
user@host> show security idp sensor-configuration
  application-identification {
    max-tcp-session-packet-memory 5000;
  }
  flow {
    drop-on-limit;
    drop-on-failover;
    drop-if-no-policy-loaded;
    max-sessions-offset 5;
    min-objcache-limit-lt 21;
    min-objcache-limit-ut 56;
  }
}
```

Meaning The **show security idp sensor-configuration** command displays all sensor configuration options that are set with certain values.

Verifying IDP Counters

Purpose Verify the IDP counters.

From operational mode, enter the **show security idp counters flow** command.

Sample Output

```
IDP counters:

  IDP counter type                                                      Value
 Fast-path packets                                                       0
 Slow-path packets                                                       0
 Session construction failed                                             0
 Session limit reached                                                   0
 Session inspection depth reached                                        0
 Memory limit reached                                                    0
 Not a new session                                                       0
 Invalid index at ageout                                                 0
 Packet logging                                                          0
 Policy cache hits                                                       0
 Policy cache misses                                                     0
 Policy cache entries                                                    0
 Maximum flow hash collisions                                            0
 Flow hash collisions                                                    0
 Gates added                                                             0
 Gate matches                                                            0
 Sessions deleted                                                        0
 Sessions aged-out                                                       0
 Sessions in-use while aged-out                                          0
 TCP flows marked dead on RST/FIN                                        0
 Policy init failed                                                      0
 Number of times Sessions exceed high mark                               0
 Number of times Sessions drop below low mark                            0
 Memory of Sessions exceeds high mark                                    0
 Memory of Sessions drops below low mark                                 0
 SM Sessions encountered memory failures                                 0
 SM Packets on sessions with  memory failures                            0
 IDP session gate creation requests                                      0
 IDP session gate creation acknowledgements                              0
 IDP session gate hits                                                   0
 IDP session gate timeouts                                               0
 Number of times Sessions crossed the CPU threshold value that is set    0
 Number of times Sessions crossed the CPU upper threshold                0
 Sessions constructed                                                    0
 SM Sessions ignored                                                     0
 SM Sessions dropped                                                     0
 SM Sessions interested                                                  0
 SM Sessions not interested                                              749
 SM Sessions interest error                                              0
 Sessions destructed                                                     0
 SM Session Create                                                       0
 SM Packet Process                                                       0
 SM ftp data session ignored by idp                                      0
 SM Session close                                                        0
 SM Client-to-server packets                                             0
 SM Server-to-client packets                                             0
 SM Client-to-server L7 bytes                                            0
 SM Server-to-client L7 bytes                                            0
 Client-to-server flows ignored                                          0
 Server-to-client flows ignored                                          0
 Both directions flows ignored                                           0
 Fail-over sessions dropped                                              0
 Sessions dropped due to no policy                                       0
 IDP Stream Sessions dropped due to memory failure                       0
 IDP Stream Sessions ignored due to memory failure                      0
 IDP Stream Sessions closed due to memory failure                       0
 IDP Stream Sessions accepted                                           0
 IDP Stream Sessions constructed                                        0
 IDP Stream Sessions destructed                                         0
 IDP Stream Move Data                                                   0
 IDP Stream Sessions ignored on JSF SSL Event                           0
 IDP Stream Sessions not processed for no matching rules                0
 IDP Stream stbuf dropped                                               0
 IDP Stream stbuf reinjected                                            0
 Busy pkts from stream plugin                                           0
 Busy pkts from pkt plugin                                              0
 bad kpp                                                                0
 Lsys policy id lookup failed sessions                                  0
 Busy packets                                                           0
 Busy packet Errors                                                     0
 Dropped queued packets (async mode)                                    0
 Dropped queued packets failed(async mode)                              0
 Reinjected packets (async mode)                                        0
 Reinjected packets failed(async mode)                                  0
 AI saved processed packet                                              0
 AI-session dropped due to malloc failure before session create         0
 AI-Sessions dropped due to malloc failure after create                 0
 AI-Packets received on sessions marked for drop due to malloc failure  0
 busy packet count incremented                                          0
 busy packet count decremented                                          0
 session destructed in pme                                              0
 session destruct set in pme                                            0
 kq op hold                                                             0
 kq op drop                                                             0
 kq op route                                                            0
 kq op continue                                                         0
 kq op error                                                            0
 kq op stop                                                             0
 PME wait not set                                                       0
 PME wait set                                                           0
 PME KQ run not called                                                  0
```

Meaning The **show security idp counters flow** command displays all counters that are used for analyzing dropped failover traffic, dropped IDP traffic, and dropped application identification traffic.

See Also

- sensor-configuration on page 465

Release History Table

Release	Description
12.3X48-D10	Starting with Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, IDP Intelligent Bypass feature is supported on SRX Series.

Related Documentation

- IDP Policies Overview on page 43
- IDP Policy Rules and IDP Rule Bases on page 55

IDP Security Packet Capture

Supported Platforms SRX Series, vSRX

An IDP sensor configuration defines the device specifications for the packet capture.

For more information, see the following topics:

- Understanding Security Packet Capture on page 273
- Example: Configuring Security Packet Capture on page 274
- Example: Configuring Packet Capture for Datapath Debugging on page 277

Understanding Security Packet Capture

Viewing packets that precede and follow an attack helps you determine the purpose and extent of an attempted attack, whether an attack was successful, and if any network damage was caused by an attack. Packet analysis also aids in defining attack signatures to minimize false positives.

If packet capture is enabled when an attack is logged, a specified number of packets before and after the attack can be captured for the session. When all packets have been collected, they are transmitted in Device Management Interface (DMI) to a host device for offline analysis.

A notification option in the IDP policy rule enables packet capture when a rule match occurs. The option further defines the number of packets to be captured and the duration of packet capture for the associated session.

An IDP sensor configuration defines the device specifications for the packet capture. Options for this command determine the memory to be allocated for packet capture, and the source and host devices between which the packet capture object will be transmitted.

A **show** command displays packet capture counters that provide details about the progress, success, and failure of packet capture activity on the device.

Support for packet capture is available only once on each session.

NOTE: **When packet capturing is configured with an improved pre-attack configuration parameter value, the resource usage increases proportionally and might affect the performance of your device.**

See Also

- Understanding IDP Logging on page 257

Example: Configuring Security Packet Capture

This example shows how to configure the security packet capture.

- Requirements on page 274
- Overview on page 274
- Configuration on page 274
- Verification on page 276

Requirements

Before you begin, configure network interfaces.

Overview

In this example, you configure a packet capture for rule 1 of policy pol0. The rule specifies that, if an attack occurs, 10 packets before the attack and 3 packets after the attack will be captured, and that the post-attack capture should time out after 60 seconds. The sensor configuration is modified to allocate 5 percent of available memory and 15 percent of the IDP sessions to packet capture. When the packet capture object is prepared, it is transmitted from device 10.56.97.3 to port 5 on device 10.24.45.7.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security idp idp-policy pol0 rulebase-ips rule 1 then notification packet-log pre-attack
  10 post-attack 3 post-attack-timeout 60
set security idp sensor-configuration packet-log total-memory 5 max-sessions 15
  source-address 10.56.97.3 host 10.24.45.7 port 5
set security idp sensor-configuration log suppression disable
set security idp idp-policy pol0 rulebase-ips rule 1 match attacks predefined-attack-groups
  "TELNET-Critical"
set security idp idp-policy pol0 rulebase-ips rule 1 then action drop-packet
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the security packet capture:

1. Create an IDP policy.

```
[edit]
user@host# edit security idp idp-policy pol0
```

2. Associate a rulebase with the policy.

```
[edit edit security idp idp-policy pol0]
```

```
user@host# edit rulebase-ips
```

3. Add rules to the rulebase.

   ```
   [edit edit security idp idp-policy pol0 rulebase-ips]
   user@host# edit rule 1
   ```

4. Specify notification, define the size and timing constraints for each packet capture.

   ```
   [edit security idp idp-policy pol0 rulebase-ips rule 1 ]
   user@host# set then notification packet-log pre-attack 10 post-attack 3
     post-attack-timeout 60
   ```

5. Define an attack as match criteria.

   ```
   [edit security idp idp-policy pol0 rulebase-ips rule 1]
   user@host# set match attacks predefined-attack-groups "TELNET-Critical"
   ```

6. Specify an action for the rule.

   ```
   [edit security idp idp-policy pol0 rulebase-ips rule 1]
   user@host# set then action drop-packet
   ```

7. Enable the security idp sensor-configuration.

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

8. (Optional) Disable security idp sensor-configuration log suppression.

   ```
   [edit]
   user@host# set security idp sensor-configuration log suppression disable
   ```

   NOTE: When IDP log suppression is enabled (which is the default behaviour), during incidents of high volume or repetitive attacks matching a single signature, a packet capture (PCAP) may not be generated by the SRX Series device and forwarded to the collector. It is recommended to disable IDP log suppression if you require PCAP records for each attack.

9. Allocate the device resources to be used for packet capture.

   ```
   [edit security idp sensor-configuration]
   user@host# set packet-log total-memory 5 max-sessions 15
   ```

10. Identify the source and host devices for transmitting the packet-capture object.

    ```
    [edit security idp sensor-configuration]
    user@host# set packet-log source-address 10.56.97.3 host 10.24.45.7 port 5
    ```

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security idp

idp-policy pol0 {
  rulebase-ips {
    rule 1 {
      match {
        attacks {
          predefined-attack-groups TELNET-Critical;
        }
      }
      then {
        action {
          drop-packet;
        }
        notification {
          packet-log {
            pre-attack 10;
            post-attack 3;
            post-attack-timeout 60;
          }
        }
      }
    }
  }
}

sensor-configuration {
  log {
    suppression {
      disable;
    }
  }
  packet-log {
    total-memory 5;
    max-sessions 15;
    source-address 10.56.97.3;
    host {
      10.24.45.7;
      port 5;
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verifying Security Packet Capture on page 277

Verifying Security Packet Capture

Purpose Verify security packet capture.

Action From operational mode, enter the **show security idp counters packet-log** command.

user@host> **show security idp counters packet-log**

```
IDP counters:                                                       Value
 Total packets captured since packet capture was activated               0
 Total sessions enabled since packet capture was activated               0
 Sessions currently enabled for packet capture                           0
 Packets currently captured for enabled sessions                         0
 Packet clone failures                                                   0
 Session log object failures                                             0
 Session packet log object failures                                      0
 Sessions skipped because session limit exceeded                         0
 Packets skipped because packet limit exceeded                           0
 Packets skipped because total memory limit exceeded                     0
```

Example: Configuring Packet Capture for Datapath Debugging

This example shows how to configure packet capture to monitor traffic that passes through the device. Packet capture then dumps the packets into a PCAP file format that can be later examined by the tcpdump utility.

- Requirements on page 277
- Overview on page 277
- Configuration on page 278
- Verification on page 279

Requirements

Before you begin, see *Debugging the Data Path (CLI Procedure)*.

Overview

A filter is defined to filter traffic; then an action profile is applied to the filtered traffic. The action profile specifies a variety of actions on the processing unit. One of the supported actions is packet dump, which sends the packet to the Routing Engine and stores it in proprietary form to be read using the **show security datapath-debug capture** command.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security datapath-debug capture-file my-capture
set security datapath-debug capture-file format pcap
set security datapath-debug capture-file size 1m
set security datapath-debug capture-file files 5
set security datapath-debug maximum-capture-size 400
set security datapath-debug action-profile do-capture event np-ingress packet-dump
set security datapath-debug packet-filter my-filter action-profile do-capture
set security datapath-debug packet-filter my-filter source-prefix 1.2.3.4/32
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*in the *CLI User Guide*.

To configure packet capture:

1. Edit the security datapath-debug option for the multiple processing units along the packet-processing path:

   ```
   [edit]
   user@host# edit security datapath-debug
   ```

2. Enable the capture file, the file format, the file size, and the number of files. Size number limits the size of the capture file. After the limit size is reached, if the file number is specified, then the capture file will be rotated to filename *x*, where *x* is auto-incremented until it reaches the specified index and then returns to zero. If no files index is specified, the packets will be discarded after the size limit is reached. The default size is 512 kilobytes.

   ```
   [edit security datapath-debug]
   user@host# set capture-file my-capture format pcap size 1m files 5
   [edit security datapath-debug]
   user@host# set maximum-capture-size 400
   ```

3. Enable action profile and set the event. Set the action profile as do-capture and the event type as np-ingress:

   ```
   [edit security datapath-debug]
   user@host# edit action-profile do-capture
   [edit security datapath-debug action-profile do-capture]
   user@host# edit event np-ingress
   ```

4. Enable packet dump for the action profile:

   ```
   [edit security datapath-debug action-profile do-capture event np-ingress]
   user@host# set packet-dump
   ```

5. Enable packet filter, action, and filter options. The packet filter is set to my-filter, the action profile is set to do-capture, and filter option is set to source-prefix 1.2.3.4/32.

```
[edit security datapath-debug]
user@host# set security datapath-debug packet-filter my-filter action-profile
  do-capture
```

```
[edit security datapath-debug]
user@host#  set security datapath-debug packet-filter my-filter source-prefix
  1.2.3.4/32
```

Results From configuration mode, confirm your configuration by entering the **show security datapath-debug** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it. The following is**show security datapath-debug** output from the **show security datapath-debug** command:

```
security {
  datapath-debug {
    capture-file {
      my-capture
      format pcap
      size 1m
      files 5;
    }
  }
  maximum-capture-size 100;
  action-profile do-capture {
    event np-ingress {
      packet-dump
    }
  }
  packet-filter my-filter {
    source-prefix 1.2.3.4/32
    action-profile do-capture
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verifying Packet Capture on page 279
- Verifying Data Path Debugging Capture on page 280
- Verifying Data Path Debugging Counter on page 280

Verifying Packet Capture

Purpose Verify if the packet capture is working.

Action From operational mode, enter the **request security datapath-debug capture start** command to start packet capture and enter the **request security datapath-debug capture stop** command to stop packet capture.

To view the results, from CLI operational mode, access the local UNIX shell and navigate to the directory /var/log/my-capture. The result can be read by using the tcpdump utility.

Verifying Data Path Debugging Capture

Purpose Verify the details of data path debugging capture file.

Action From operational mode, enter the **show security datapath-debug capture** command.

```
user@host>show security datapath-debug capture
```

WARNING: When you are done troubleshooting, make sure to remove or deactivate all the traceoptions configurations (not limited to flow traceoptions) and the complete security datapath-debug configuration stanza. If any debugging configurations remain active, they will continue to use the device's CPU and memory resources.

Verifying Data Path Debugging Counter

Purpose Verify the details of the data path debugging counter.

Action From operational mode, enter the **show security datapath-debug counter** command.

IDP Performance and Capacity Tuning

Supported Platforms SRX Series, vSRX

This topic provides an overview on performance and capacity tuning for an Intrusion Detection and Prevention (IDP) session.

For more information, see the following topics:

- Performance and Capacity Tuning for IDP Overview on page 280
- Configuring Session Capacity for IDP (CLI Procedure) on page 281

Performance and Capacity Tuning for IDP Overview

This topic provides an overview on performance and capacity tuning for an Intrusion Detection and Prevention (IDP) session.

If you are deploying IDP policies, you can configure the device to increase IDP session capacity. By using the provided commands to change the way the system allocates resources, you can achieve higher IDP session capacity.

By using the **maximize-idp-sessions** command, you can increase the IDP session capacity. In this mode, by default, the device assigns a greater weight value to firewall functions. Based on your IDP policy, you can shift the weight to IDP functions to maximize IDP performance. By shifting weight, you are increasing capacity and allocating more processing power for the given service.

NOTE: **You should not configure the device to increase IDP session capacity if you are not using an IDP policy.**

The device ships with an implicit default session capacity setting. This default value adds weight to firewall sessions. You can manually override the default by adding the **maximize-idp-sessions** setting to your configuration. When you do this, in addition to IDP session scaling, you can choose to assign weight values of equal, firewall, or IDP to firewall and IDP functions. Typically, when you only include IDP-recommended attacks or client-to-server attacks in your IDP policy, IDP functions consume less CPU resources, for this reason, you would select weight firewall to maximize device performance. Alternatively, if you add server-to-client attacks to your IDP policy, IDP functions consume higher CPU resources. For this reason, you would select weight IDP to maximize performance. Essentially, you will need to configure the weight based on the desired IDP policy and performance. You do this by examining the CPU resource utilization on the packet forwarding engine by using the **show security monitoring fpc *number*** command.

See Also

- IDP Policies Overview on page 43

Configuring Session Capacity for IDP (CLI Procedure)

The configuration instructions in this topic describe how modify session capacity for IDP policies.

You do this by adding the **maximize-idp-sessions** command and then adding the weight option to specify IDP sessions.

NOTE: **The weight option depends on the maximize-idp-sessions command being set.**

1. If you have an active IDP policy, you can configure the device to increase IDP session capacity by entering following command:

   ```
   user@host# set security forwarding-process application-services maximize-idp-sessions
   ```

2. You can further adjust the weight of the firewall and IDP processing functions, such as in the case of heavier IDP policies with the following command:.

```
user@host# set security forwarding-process application-services maximize-idp-sessions
  weight idp
```

3. Commit your changes. You must reboot the device for any session capacity setting changes to take effect.

   NOTE: If the device has **maximize-idp-sessions** weight enabled for IDP, and you do not have an IDP policy configured, a warning message appears when you commit your configuration. If you see this warning, you should remove your configured settings.

To turn **maximize-idp-sessions** settings off, remove the **maximize-idp-sessions** configuration.

NOTE: You must reboot the device for any **maximize-idp-sessions** setting changes to take effect.

See Also

- IDP Policies Overview on page 43

CHAPTER 6

Migrating from IDP Series or ISG Series Devices to SRX Series Devices

Introduction to IDP Migration

Supported Platforms SRX Series, vSRX

This topic provides a brief overview of some basic considerations when moving from standalone Juniper Networks IDP Series Intrusion Detection and Protection Appliances or Juniper Networks ISG Series Integrated Security Gateways with IDP security module to the Juniper Networks SRX Series Services Gateways.

For more information, see the following topics:

IDP Series Appliances to SRX Series Devices Migration Overview

Introduction

SRX Series devices are equipped with full security and networking capabilities and represents the highest performing firewalls with natively integrated full intrusion prevention system (IPS) technology from Juniper Networks IDP Series Intrusion Detection and Prevention Appliances, providing inline protection against current and emerging threats throughout the network.

Although an SRX Series IDP policy can be configured entirely from within Juniper Networks J-Web software, this document focuses primarily on the CLI and Junos Space Security Director configuration steps, with the intention of providing an easy transition and learning path for both system engineers new to the IDP Series and those already familiar with managing standalone IDP Series and ISG Series with IDP solutions.

Because standalone IDP Series devices are typically deployed in either sniffer or transparent mode, additional considerations regarding network design must be addressed. These involve:

- Network interfaces configuration
- Security zones configuration

In addition, there are considerations regarding the following security features:

- Denial of service (DoS) and flood protection.
- Traffic anomaly detection or screens (as well as some of the detection methods applicable for SRX Series devices).
- Configured settings and actions must be closely analyzed because adding a new device can potentially impact network traffic—particularly in regard to Layer 3 processing.

SRX Series Services Gateways can be deployed in sniffer mode (only on SRX5400, SRX5600, and SRX5800 devices). The sniffer mode is not supported on SRX300, SRX340, SRX345, and SRX550HM devices.

Multimethod Detection

SRX Series devices deploy two rulebases—a main IDP rulebase and an exempt rulebase.

In addition, SRX Series devices use security zones that are based on technology available with ScreenOS-based security devices, and provide detailed screen protection as an alternative for some basic standalone detection methods or rulebases.

Logging

Logging on an SRX Series device must be configured to send records in response to security events through system logging to a preconfigured syslog server, such as the Juniper Networks Juniper Secure Analytics (JSA).

Sensor Configuration Settings

On both standalone IDP Series and SRX Series devices, a number of sensor configuration settings can be configured to fine-tune IDP Series behavior and can be accessed from the CLI and Junos Space Security Director (SD). If any of the settings have been changed from the default value or need to be further modified, you must manually modify them. There are no automated processes to export or import modified settings.

Key Points to Consider

Note the following key points when you migrate from IDP Series Appliances to SRX Series devices:

- In comparison with deep inspection on ScreenOS, the fundamental IPS detection capabilities on the SRX Series devices do not differ from that available on IDP Series Appliances or ISG Series with IDP security modules.
- Not all IPS features are available on SRX Series IDP. We recommend that you familiarize yourself with documentation that details those differences.
- Only SRX5400, SRX5600, and SRX5800 devices can be configured in sniffer mode (transparent mode).
- IPS does not need a separate license to run as a service on the SRX Series device; however, a license is required for IPS updates.
- A base firewall policy is required and needs to include an IPS application-service statement to enable IPS inspection.
- Enabling all attacks is not supported. If the policy does not load, check the service log files for policy size and load results.
- A system log (syslog) server is required to collect security event-related messages when the messages are identified on the SRX Series data plane.
- It is s important to understand that compiling and applying an IPS policy can take some time, depending on the number of attack objects and the size of the policy. Starting with Junos OS Release 12.1 and Junos OS Release 17.3R1, SRX Series devices are leveraged for smarter compilation engine along with caching compiled information so that the compilation process takes much less time. The compilation process is conducted asynchronously, which means that the SRX Series device starts the process but will not hold up CLI or SD session, but instead will allow you to check back later on the status.

See Also

- Installing the IPS License (CLI) on page 289

Understanding Intrusion Prevention System for SRX Series Devices

Overview

The Juniper Networks intrusion prevention system (IPS) feature detects and prevents attacks in network traffic.

SRX Series devices provide the IPS functionality integrated within the Junos OS software; no special hardware is needed. IPS administrators have the option of deploying and administering IPS using the CLI or the Junos Space Security Director.

IPS Architecture

The IPS architecture is composed of the following:

- SRX Series device with IPS—IPS functionality is integrated as part of Junos OS and no special hardware is required.
- Management—SRX Series devices can be fully managed using the CLI commands. However, if there are multiple SRX Series devices involved in the IPS deployment, we recommend using the Junos Space Security Director application.
- Logging—Juniper Secure Analytics (JSA) is Juniper Networks' security information and event management (SIEM) solution. JSA has predefined dashboards and reports for the SRX Series devices IPS solution. In addition to logging, JSA provides event correlation, incident management, and flow monitoring. SRX Series logs are in syslog (structured data syslog) format, and these can be sent to JSA or to any other syslog servers that users might already have in place.

IPS with Chassis Clustering Limitations

IPS is supported in both active/passive and active/active chassis cluster modes on SRX Series devices with the following limitations:

- No inspection is performed on sessions that fail over or fail back. Only new sessions after a failover are inspected by IPS, and older sessions become firewall sessions.
- The IP action table is not synchronized across nodes. If an IP action is taken for a session, and the source IP, destination IP or both is added to the IP action table, this information is not synchronized to the secondary node. Therefore, the sessions from the source IP, destination IP or both will be forwarded until a new attack is detected.
- The SSL session ID cache is not synchronized across nodes. If an SSL session reuses a session ID and it happens to be processed on a node other than the one on which the session ID is cached, the SSL session cannot be decrypted and will be bypassed for IPS inspection.

See Also

- IDP Series Appliances to SRX Series Devices Migration Overview on page 283

Understanding the Intrusion Prevention System Deployment Modes for SRX Series Devices

This topic provide you an overview of the different types of IPS deployment modes for SRX Series devices.

IPS provides three different modes of deployment:

- Integrated mode
- Inline-tap mode
- Sniffer mode

Integrated Mode

Integrated mode is supported on SRX Series devices. Integrated mode is the default mode in which IPS operates on the SRX Series devices (There are no specific indications that show that the device is in integrated mode.)

NOTE: We recommend deploying IPS in integrated mode.

Inline-Tap Mode

Junos OS Release 10.2 and later supports inline-tap mode only on SRX5400, SRX5600, and SRX5800 devices.

The main purpose of inline-tap mode is to provide best-case deep inspection analysis of traffic while maintaining overall performance and stability of the device. When a device is in inline-tap mode, the firewall process (flowd) processes the firewall traffic as normal, but makes a copy of the packet and puts it in a queue for the independent IPS module (idpd) to inspect. In the meantime, flowd forwards the original packet without waiting for idpd to perform the inspection.

Because inline tap mode puts IPS in a passive mode for inspection, preventative actions such as close, drop, and mark diffserv are deferred. The drop packet action is ignored.

NOTE: In inline-tap mode, the SRX Series device with IPS provides minimum protection. Upon detecting an attack, idpd can reset a session, but by the time the reset occurs, flowd would have allowed malicious packets through the network.

Sniffer Mode

Sniffer mode is supported only on SRX5400, SRX5600, and SRX5800 devices. You can use the sniffer mode of IPS deployment by configuring the interfaces in promiscuous mode and manipulating the traffic and flow setup with routing.

On SRX5400, SRX5600, and SRX5800 devices, in sniffer mode, ingress and egress interfaces work with flow showing both source and destination interface as egress interface.

As a workaround, in sniffer mode, use the tagged interfaces. Hence, the same interface names are displayed in the logs. For example, ge-0/0/2.0 as ingress (sniffer) interface and ge-0/0/2.100 as egress interface are displayed in the logs to show the source interface as ge-0/0/2.100.

```
set interfaces ge-0/0/2 promiscuous-mode
set interfaces ge-0/0/2 vlan-tagging
set interfaces ge-0/0/2 unit 0 vlan-id 0
set interfaces ge-0/0/2 unit 100 vlan-id 100
```

See Also

- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Installing the IPS License (CLI) on page 289

Getting Started with IPS on SRX Series Devices

Before configuring the SRX Series device for IPS functionality, perform the following tasks:

1. **Install the License**—You must install an IDP license before you can download any attack objects. If you are using only custom attack objects, you do not need to install a license, but if you want to download Juniper Networks predefined attack objects, you must have this license. Juniper provides you with the ability to download a 30-day trial license to permit this functionality for a brief period of time to evaluate the functionality. All you need is run the **request system license add** command either specifying a file storage location or copy and paste it into the terminal.
2. **Configure Network Access**—Before you can download the attack objects, you must have network connectivity to either the Juniper download server or a local server from which the signatures can be downloaded. This typically requires network configuration (IP/Netmask, routing, and DNS) and permitted access to reach the server. At the time of this writing, HTTP proxies are not supported, but you can configure a local webserver from which to serve the files.
3. **Download Attack Objects**—Before deploying the IPS, you must first download the attack objects from which the policy will be compiled. Triggering a manual download does not configure the SRX Series device to download them in the future, so you must configure automatic updates to download them.
4. **Install Attack Objects**—Once the download has been completed, you must install the attack updates before they are actually used in a policy. If you already have a policy configured, you do not need to recommit the policy—installing the updates adds them to the policy. The installation process compiles the attack objects that have been downloaded to a stage directory into the configured policy.
5. **Download Policy Templates (optional)**—You can optionally download and install predefined IPS policies known as policy templates provided by Juniper to get started. After finishing this chapter, you should be able to configure your own policy, so you probably will not need policy templates.

NOTE: Starting with Junos OS Release 12.1 and Junos OS Release 17.3R1, the SRX Series devices automatically push the signature package to the secondary member of the chassis cluster. Prior to Junos OS Release 12.1 and Junos OS Release 17.3R1, you had to use the fxp0 on both members of the cluster because both members had to download their own instance. With Junos OS Releases beyond 12.1 and 17.3R1, there is no explicit configuration. SRX Series device will download the signature package and push it to the secondary member during the download process.

Related Documentation

- Understanding IDP Signature Database for Migration on page 298

Understanding IDP Migration

Supported Platforms SRX Series, vSRX

This topic provides details on installing and configuring IDP.

For more information, see the following topics:

- Installing the IPS License (CLI) on page 289
- Initial Configuration Overview on page 290
- IPS Configuration (CLI) on page 291

Installing the IPS License (CLI)

You can either download an IPS license from the license server, or manually install IPS if you received a license from Juniper Networks.

Access the SRX Series device console through the serial cable plugged into the console port on the device or by using a terminal session such as SSH.

To apply your IPS subscription license to the device, use the following CLI command:

```
user@host> request system license update
```

If you received a license for manual installation, perform the following tasks:

1. Access the SRX Series Services Gateway console either by plugging the serial cable into the console port on the device or by using a terminal session such as SSH.

2. Check for an IPS license (required for all IPS updates):

   ```
   user@host> show system license
   ```

   ```
   License usage:
                                    Licenses     Licenses    Licenses    Expiry
     Feature name                       used    installed      needed
     logical-system                        0            0           0    permanent
   ```

3. If there are no licenses installed, obtain the chassis serial number by using the following CLI command:

   ```
   user@host> show chassis hardware
   ```

4. A serial number is needed to generate the IPS license. You can add a license key from a file or URL or from the terminal.

   - From a file or URL:

     ```
     user@host> request system license add <file name>
     ```

   - From the terminal:

```
user@host> request system license add terminal
```

5. When prompted, enter the license key, separating multiple license keys with a blank line. If the license key you enter is invalid, an error is generated when you press Ctrl-D to exit license entry mode.

6. Verify the system license by entering the **show system license** command.

```
user@host> show system license
```

```
License usage:
                   Licenses     Licenses    Licenses   Expiry
  Feature name         used    installed      needed
  idp-sig                4            1           0  permanent


Licenses installed:
  License identifier: JUNOS208639
  License version: 2
  Valid for device: AA4508AD0005
  Features:
    idp-sig   - IDP Signature
     date-based, 2009-0406 08:00:00 GMT-8 - 2010-04-06 08:00:00 GMT-8
```

See Also

- *Understanding Chassis Cluster Licensing Requirements*
- *Verifying Licenses on an SRX Series Device in a Chassis Cluster*
- *Installing Licenses on the SRX Series Devices in a Chassis Cluster*
- *Understanding Licenses for Logical Systems on SRX Series Devices*

Initial Configuration Overview

Enabling a fully functional IPS service on SRX Series Services Gateways includes the following basic configuration steps:

Basic Configurations

1. Configure basic networking, security, and access components (in most cases this will already be configured).

2. Configure and activate IPS policy.

3. Configure firewall policy to associate specific rules with IPS.

4. Download attack objects including sensor updates.

5. Configure logging.

6. Update security-package.

7. Verify configuration and test functionality.

Initial Configuration Assumptions

Before starting the IPS policy configuration, this document assumes that an initial networking configuration exists and that an admin user has full access to the SRX Series. Initial device configuration on our sample system is as follows:

```
user@ost > show configuration | display set
set system root-authentication encrypted-password "$ABC123"
set system name-server 1.2.3.4
set system login user mxb uid 2000
set system login user mxb class super-user
set system login user mxb authentication encrypted-password "$123ABC"
set system syslog user * any emergency
set system syslog file messages any notice
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval
set interfaces fxp0 unit 0 family inet address 192.168.1.221/24
set routing-options static route 0.0.0.0/0 next-hop 192.168.1.1
set security idp security-package url https://services.netscreen.com/cgi-bin/index.cgi
```

NOTE: Throughout this document we provide commands required to configure specific features; however, in order to activate associated functionality, configuration changes need to be successfully committed (using the commit command).

See Also

- IDP Series Appliances to SRX Series Devices Migration Overview on page 283
- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Understanding the Intrusion Prevention System Deployment Modes for SRX Series Devices on page 286

IPS Configuration (CLI)

- Configuring Interfaces on page 291
- Configuring Security Zones on page 292
- Configuring IPS Security Policy on page 294
- Configuring Firewall Security Policy on page 296
- IPS Logging on page 297

Configuring Interfaces

1. Display current interfaces (assumption is interfaces have been properly cabled)

```
user@host# configure
```

```
fxp0 {
  unit 0 {
    family inet {
      address 192.168.1.221/24;
    }
  }
}
[edit]
user@host# run show interfaces | match ge-0/0
Physical interface: ge-0/0/0, Enabled, Physical link is Down
Physical interface: ge-0/0/1, Enabled, Physical link is Down
Physical interface: ge-0/0/2, Enabled, Physical link is Up
Physical interface: ge-0/0/3, Enabled, Physical link is Up
Physical interface: ge-0/0/4, Enabled, Physical link is Down
Physical interface: ge-0/0/5, Enabled, Physical link is Down
Physical interface: ge-0/0/6, Enabled, Physical link is Down
Physical interface: ge-0/0/7, Enabled, Physical link is Up
Physical interface: ge-0/0/8, Enabled, Physical link is Down
Physical interface: ge-0/0/9, Enabled, Physical link is Down
Physical interface: ge-0/0/10, Enabled, Physical link is Down
Physical interface: ge-0/0/11, Enabled, Physical link is Down
```

2. Configure forwarding interfaces.

   ```
   user@host# set interfaces ge-0/0/2 unit 0 family inet address 33.3.3.1/24
   user@host# set interfaces ge-0/0/3 unit 0 family inet address 44.4.4.1/24
   ```

3. Verify the configuration.

   ```
   user@host# run show interfaces terse | match /24
   ```

   ```
   ge-0/0/2.0  up up inet 33.3.3.1/24
       ge-0/0/3.0 up up inet 44.4.4.1/24
        ge-0/0/7.0 up up inet 192.168.2.222/24
       fxp0.0     up up inet 192.168.1.221/24
   ```

Configuring Security Zones

1. Configure security zones.

   a. Display existing zones:

      ```
      user@host> show security zones
      ```

      ```
      Security zone: junos-global
      Send reset for non-SYN session TCP packets: Off
      Policy configurable: Yes
      Interfaces bound: 0
      Interfaces:
      ```

   b. Configure zones abc-trust and abc-untrust and assign interfaces accordingly.

      ```
      user@host# set security zones security-zone abc-trust interfaces ge-0/0/2
      user@host# set security zones security-zone abc-untrust interfaces ge-0/0/3
      ```

2. Verify the configuration.

```
user@host# run show security zones
```

```
Security zone: abc-trust
Send reset for non-SYN session TCP packets: Off
Policy configurable: Yes
Interfaces bound: 1
Interfaces:
ge-0/0/2.0

Security zone: abc-untrust
Send reset for non-SYN session TCP packets: Off
Policy configurable: Yes
Interfaces bound: 1
Interfaces:
ge-0/0/3.0

Security zone: junos-global
Send reset for non-SYN session TCP packets: Off
Policy configurable: Yes
Interfaces bound: 0
Interfaces:
```

Configuring IPS Security Policy

1. Configure IPS policy abc-idp-policy.

   The simple configuration in this section involves setting up one rule looking for all critical attacks and, in case a match is found, dropping the associated connection, setting that event as critical and logging it with an alert. The second rule is configured to look for major attacks and to perform a recommended action upon detecting a severe attack, as well as logging the event.

   NOTE: Logging means sending a system log (syslog) message to an appropriate, preconfigured syslog server. Logging configuration steps are provided in subsequent sections.

```
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 1
  match from-zone any to-zone any source-address any destination-address
  any application any attacks predefined-attack-groups Critical
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 1
  then severity critical notification log-attacks alert
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 1
  match from-zone any to-zone any source-address any destination-except
  address
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 1
  match from-zone any to-zone any source-address any source-except
  address
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 2
  match from-zone any to-zone any source-address any destination-address
  any application any attacks predefined-attack-groups Major
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 2
  then action recommended
user@host# set security idp idp-policy abc-idp-policy rulebase-ips rule 2
  then severity major notification log-attacks
```

2. Verify IPS policy abc-idp-policy.

```
user@host> show security idp idp-policy abc-idp-policy
rulebase-ips {
  rule 1 {
    match {
      from-zone any;
      source-except address;
      to-zone any;
      destination-except address;
      application any;
      attacks {
        predefined-attack-groups Critical;
      }
    }
    then {
      notification {
        log-attacks {
          alert;
        }
      }
      severity critical;
    }
  }
  rule 2 {
    match {
      from-zone any;
      source-address any;
      to-zone any;
      destination-address any;
      application any;
      attacks {
        predefined-attack-groups Major;
      }
    }
    then {
      action {
        recommended;
      }
      notification {
        log-attacks;
      }
      severity major;
    }
  }
}
```

3. Set trace options.

   a. To provide detailed IPS process event information (policy compilation result, policy loading results, dfa matches, and so on) which allows for further system analysis, tuning, and easier troubleshooting, it is highly recommended to enable trace options. The following is an example setting that configures trace to write all security events encompassing all debug levels (error, info, notice, verbose, and warning). The trace filename is not specified trace if it is not written into the file named after the process being traced, which is the case with IDP/var/log/idpd:

      ```
      user@host# set security idp traceoptions flag all
      user@host# set security idp traceoptions level all
      ```

   b. For this example, we limit the file size to 100 MB. This means that the process will write this file and once it reaches 100 MB, it will rename it to idpd.0 and continue with a new idpd. The default number of files is 3 and if file numbers are exhausted, the oldest file (idpd.2) gets overwritten.

      ```
      user@host# set security idp traceoptions file size 100M
      ```

4. Verify trace options settings.

   ```
   user@host> show security idp traceoptions
   ```

   ```
   file size 100m;
   flag all;
   ```

```
level all;
no-remote-trace;
```

5. Activate IPS Series policy.

   ```
   user@host# set security idp active-policy abc-idp-policy
   ```

6. Verify active IPS policy.

   ```
   user@host> show security idp active-policy
   ```

   ```
   active-policy abc-idp-policy;
   ```

NOTE: To deploy IPS policy on the SRX Series devices, one more step is required—configuring firewall security policy to identify which traffic is to be processed by the IPS service. This is described in the following section.

Configuring Firewall Security Policy

For traffic entering the SRX Series device to be processed by IPS security policy firewall, the security policy needs to be configured accordingly.

Following are steps required to configure firewall security policy and finalize Intrusion Prevention System configuration on the SRX Series gateway. This will result in traffic between security zones abc-untrust and abc-trust being inspected by IPS security policy abc-idp-policy.

1. Ensure that the system is configured with the default policy denying all traffic. This basically means traffic will 1. be denied throughout the gateway unless specifically allowed to by firewall security policy.

   ```
   user@host> show security policies
   ```

   ```
   Default policy: deny-all
   ```

2. Configure policy.

   ```
   user@host# set security policies from-zone abc-untrust to-zone abc-trust policy abc
     match source-address any destination-address any application any
   user@host# security policies from-zone abc-untrust to-zone abc-trust policy abc then
     permit application-services idp
   user@host# set security policies from-zone abc-trust to-zone abc-untrust policy abc
     match source-address any destination-address any application any
   user@host# set security policies from-zone abc-trust to-zone abc-untrust policy abc
     then permit application-services idp
   ```

3. Verify configuration.

   ```
   user@host> show security policies
   from-zone abc-untrust to-zone abc-trust {
     policy abc {
       match {
         source-address any;
   ```

```
            destination-address any;
            application any;
        }
        then {
            permit {
                application-services {
                    idp;
                }
            }
        }
    }
}
from-zone abc-trust to-zone abc-untrust {
    policy abc {
        match {
            source-address any;
            destination-address any;
            application any;
        }
        then {
            permit {
                application-services {
                    idp;
                }
            }
        }
    }
}
```

IPS Logging

IPS generates event logs when an event matches an IPS policy rule in which logging is enabled. When you configure a rule for logging, the device creates a log entry for each event that matches that rule.

When configured to do so, an IPS service will send events that match policy entry to the logging server directly from the data plane via emulated IP address, encapsulated in 514/udp.

Configure logging:

1. Configure interface data plane to send syslog messages from:

   ```
   user@host# set interfaces ge-0/0/7 unit 0 family inet address 192.168.2.1/24
   ```

2. Choose the format (standard or structured format).

   ```
   user@host# set security log format syslog
   ```

3. Set the emulated source IP address (interface cannot be fxp0).

   ```
   user@host# set security log source-address 192.168.2.211
   ```

4. Set severity.

```
user@host# set security log stream jet severity debug
```

5. Indicate the syslog server IP address (to which logs are sent via 514/udp).

```
user@host# set security log stream jet host 192.168.2.212
```

6. Verify log configuration.

```
user@host> show security log
```

```
format syslog;
source-address 192.168.2.211;
stream jet {
severity debug;
host {
192.168.2.212;
}
```

See Also

- IDP Series Appliances to SRX Series Devices Migration Overview on page 283
- Installing the IPS License (CLI) on page 289
- Initial Configuration Overview on page 290

Related Documentation

- Introduction to IDP Migration on page 283

Understanding IDP Signature Database for Migration

Supported Platforms SRX Series, vSRX

The signature database is one of the major components of the intrusion prevention system (IPS). It contains definitions of different objects, such as attack objects, application signature objects, and service objects, that are used in defining IDP policy rules.

For more information, see the following topics:

- Understanding the IPS Signature Database on page 298
- Managing the IPS Signature Database (CLI) on page 300
- Managing the IPS Signature Database (Security Director) on page 304
- Example: Updating the IPS Signature Database Manually on page 306
- Example: Downloading and Installing the IPS Signature Package in Chassis Cluster Mode on page 310

Understanding the IPS Signature Database

The signature database is one of the major components of the intrusion prevention system (IPS). It contains definitions of different objects, such as attack objects, application signature objects, and service objects, that are used in defining IDP policy rules. As a response to new vulnerabilities, Juniper Networks periodically provides a file containing attack database updates on the Juniper Networks website. You can download this file to protect your network from new threats.

NOTE: IPS does not need a separate license to run as a service on the SRX Series device; however, a license is required for IPS updates. Custom attacks and custom attack groups in IDP policies can also be configured and installed even when a valid license and signature database are not installed on the device.

The IPS signature database is stored on the IPS-enabled device and contains definitions of predefined attack objects and groups. These attack objects and groups are designed to detect known attack patterns and protocol anomalies within the network traffic. The IPS signature database includes more than 5000 signatures and more than 1200 protocol anomalies.

IPS updates and application signature package updates are a separately licensed subscription service. You must install the IPS signature-database-license key on your device for downloading and installing daily signature database updates from the Juniper Networks website. The IPS signature license key does not provide grace period support.

NOTE: If you require both AppSecure and IPS features, you must install the application signature license in addition to the IPS signature-database-update license key.

The signature database comprises the following components:

- Detector engine—The IDP detector engine is a dynamic protocol decoder that includes support for decoding more than 60 protocols and more than 500 service contexts. You can download the protocol detector engine updates along with the signature database updates.
- Attack database—The attack signature database stores data definitions for attack objects and attack object groups. Attack objects comprise stateful signatures and traffic anomalies. You specify attack objects in IDP rulebase rules. New attacks are discovered daily, so it is important to keep your signature database up to date. You can download the attack database updates from the Juniper Networks website.
- Application signature database—The application signature database stores data definitions for application objects. Application objects are patterns that are used to identify applications that are running on standard or nonstandard ports.

NOTE: We recommend using the latest version of the signature database to ensure an up-to-date attack database.

See Also

- IDP Series Appliances to SRX Series Devices Migration Overview on page 283

- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Installing the IPS License (CLI) on page 289

Managing the IPS Signature Database (CLI)

This example shows how to install and schedule the signature database updates using the CLI.

- Requirements on page 300
- Overview on page 300
- Configuration on page 300
- Downloading and Installing the IPS Signature Package from an Older Junos OS Release Version to Newer Junos OS Release Version on page 302
- Verification on page 303

Requirements

Before you install the signature database updates, ensure that you have installed an IPS license key.

Overview

IPS signature database management comprises the following tasks:

- Update the signature database—Download the attack database updates available on the Juniper Networks website. New attacks are discovered daily, so it is important to keep your signature database up to date.
- Verify the signature database version—Each signature database has a different version number with the latest database having the highest number. You can use the CLI to display the signature database version.
- Update the protocol detector engine—You can download the protocol detector engine updates along with the signature database. The IPS protocol detector contains Application Layer protocol decoders. The detector is coupled with the IDP policy and is updated together. It is always needed at policy update time, even if there is no change in the detector.
- Schedule signature database updates—You can configure the IPS-enabled device to automatically update the signature database after a set interval.

Configuration

- Downloading and Installing the IPS Signature Package on page 301
- Verifying the Signature Database Version on page 301
- Scheduling the Signature Database Updates on page 302

Downloading and Installing the IPS Signature Package

Step-by-Step Procedure

New attacks are discovered daily, so it is important to keep your signature database up to date. In this example, you download and then install the latest signature package from the signature database server:

1. Download the attack database updates available on the Juniper Networks website:

   ```
   user@host>request security idp security-package download
   ```

   By default, when you download the security package, you download the following components into a Staging folder in your device: the latest version of the complete attack object groups table, the application objects table, and the updates to the IPS Detector Engine. Because the attack objects table is typically very large, by default the system only downloads updates to the attack objects table. However, you can download the complete attack objects table by using the **full-update** configuration option.

2. Check the security package download status:

   ```
   user@host>request security idp security-package download status
   ```

   On a successful download, the following message is displayed:

   ```
   Done;Successfully downloaded from
   (http://services.netscreen.com/cgi-bin/index.cgi).
   Version info:1884(Thu Mar 17 12:06:35 2011, Detector=11.4.140110223)
   ```

3. After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device. Install the security package:

   ```
   user@host>request security idp security-package install
   ```

4. Check the status of the install:

   ```
   user@host>request security idp security-package install status
   ```

   On a successful install, the following message is displayed:

   ```
   Done;Attack DB update: successful - [UpdateNumber=1884,ExportDate=Thu Mar
   17 12:06:35 2011,Detector=11.4.140110223]
    Updating control-plane with new detector: successful
    Updating data-plane with new attack or detector: successful
   ```

Verifying the Signature Database Version

Step-by-Step Procedure

Each signature database has a different version number with the latest database having the highest number.

- Use the CLI to verify the signature database version installed:

  ```
  user@host>show security idp security-package version
  ```

  The following sample output shows the version number for the signature package:

```
user@host> show security idp security-package-version
Attack database version:1883(Wed Mar 16 12:10:26 2011)
Detector version :12.6.140121210
Policy template version :N/A
```

Scheduling the Signature Database Updates

Step-by-Step Procedure

You can configure an IPS-enabled device to automatically update the signature database after a set interval. After the initial manual setup, we recommend that you schedule the signature updates so you always have protection against new vulnerabilities.

- To schedule the signature package download, from configuration mode, specify the start time and the interval for the download:

  ```
  user@host>set security idp security-package automatic interval interval start-time
    <YYYY-MM-DD.HH:MM:SS>
  ```

  For example, to set a schedule for the signature download every 72 hours, you use the following configuration:

  ```
  user@host>set security idp security-package automatic interval 72 start-time
  ```

Downloading and Installing the IPS Signature Package from an Older Junos OS Release Version to Newer Junos OS Release Version

Step-by-Step Procedure

Starting with Junos OS Release 17.3, when you upgrade from Junos OS Release 12.3X48 or 15.1X49 to Junos OS Release 17.3 or downgrade from Junos OS Release 17.3 to Junos OS Release 12.3X48 or 15.1X49, you must update the IPS signature package by downloading and installing the IPS signature package update.

NOTE: We recommend that you perform the IPS signature package update because if the previous IPS signature package download before an upgrade or a downgrade comprised an incremental or decremental update, then reinstalling of the IPS signature package, without downloading the IPS signature package again, updates the IPS signature package with only the incremental attacks from the last download and does not contain any attacks from the baseline release. Therefore, to avoid any IDP commit configuration failure, update the IPS signature package.

The following procedure shows how to download and install an IPS signature package and update the package from an older Junos OS release version to a newer Junos OS release version:

1. Perform a full update of the security package version.

   ```
   user@host>request security idp security-package download full-update
   ```

   By default, when you download the security package, you download the following components into a Staging folder in your device—the latest version of the complete attack object groups table, the application objects table, and the updates to the

IPS Detector Engine. Because the attack objects table is typically very large, by default the system downloads only updates to the attack objects table.

2. Check the security package download status.

   user@host> **request security idp security-package download status**

   On a successful download, the following message is displayed:

```
user@host # run request security idp security-package download status
Done;Successfully downloaded
from(https://signatures.juniper.net/cgi-bin/index.cgi).
Version info:2762(Tue Jul 26 22:26:57 2016 UTC, Detector=12.6.130160603)
```

3. Install the security package to update the security database with the newly downloaded updates from the Staging folder in your device.

   user@host> **request security idp security-package install**

4. Check the status of the install.

   user@host> **request security idp security-package install status**

   On a successful install, the following message is displayed:

```
user@host # run request security idp security-package install status
Done;Attack DB update : successful - [UpdateNumber=2771,ExportDate=Tue Aug
23 21:57:18 2016 UTC,Detector=12.6.130160603]
     Updating control-plane with new detector : successful
     Updating data-plane with new attack or detector : successful
```

NOTE: **When you upgrade from Junos OS Release 15.1X49 to Junos OS Release 17.3, the following warning message is displayed:**

```
WARNING: A full install of the security package is required after
 reboot.
WARNING: Please perform a full update of the security package using
WARNING:    "request security idp security-package download
full-update"
WARNING: followed by
WARNING:    "request security idp security-package install"
```

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the IPS Signature Database on page 303

Verifying the IPS Signature Database

Purpose Display the IPS signature database.

Action From operational mode, enter the **show security idp** command.

See Also

- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Understanding the IPS Signature Database on page 298
- Managing the IPS Signature Database (Security Director) on page 304

Managing the IPS Signature Database (Security Director)

This example shows how to install and schedule the signature database updates using Junos Space Security Director.

- Requirements on page 304
- Overview on page 304
- Configuration on page 304
- Verification on page 306

Requirements

This example uses the following hardware and software components:

- SRX Series device

Before you install the signature database updates, ensure that you have:

- Installed an IPS license key

Overview

The IPS signature database can be updated using either the CLI or Junos Space Security Director. SRX Series devices can be fully managed from the CLI; however, for large deployment scenarios that use multiple SRX Series devices, it is easier to manage the security package using a management platform.

Configuration

- Downloading and Installing the IPS Signature Package on page 304
- Verifying the Signature Database Version on page 305
- Scheduling the Signature Database Updates on page 306

Downloading and Installing the IPS Signature Package

Step-by-Step Procedure In this example, you download and then install the latest signature package from the signature database server:

1. Navigate to **Security Director->Downloads->Signature Database**.

   Choose the signature package listed as the latest and select **Action>Download** to download the signature package to Security Director.

   ```
   user@host>request security idp security-package download
   ```

By default, when you download the security package, you download the following components into a Staging folder in your device: the latest version of the complete attack object groups table, the application objects table, and the updates to the IPS Detector Engine. Because the attack objects table is typically very large, by default the system only downloads updates to the attack objects table. However, you can download the complete attack objects table by using the **full-update** configuration option.

2. Check the security package download status:

   user@host>**request security idp security-package download status**

   On a successful download, the following message is displayed:

   ```
   Done;Successfully downloaded from
   (http://services.netscreen.com/cgi-bin/index.cgi).
   Version info:1884(Thu Mar 17 12:06:35 2011, Detector=11.4.140110223)
   ```

3. After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device. Install the security package:

   user@host>**request security idp security-package install**

4. Check the status of the install:

   user@host>**request security idp security-package install status**

   On a successful install, the following message is displayed:

   ```
   Done;Attack DB update: successful - [UpdateNumber=1884,ExportDate=Thu Mar
   17 12:06:35 2011,Detector=11.4.140110223]
    Updating control-plane with new detector: successful
    Updating data-plane with new attack or detector: successful
   ```

Verifying the Signature Database Version

Step-by-Step Procedure

Each signature database has a different version number with the latest database having the highest number.

- Use the CLI to verify the signature database version installed:

  user@host>**show security idp security-package version**

  The following sample output shows the version number for the signature package:

  ```
  user@host> show security idp security-package-version
  Attack database version:1883(Wed Mar 16 12:10:26 2011)
  Detector version :12.6.140121210
  Policy template version :N/A
  ```

Scheduling the Signature Database Updates

Step-by-Step Procedure

You can configure IPS-enabled device to automatically update the signature database after a set interval. After the initial manual setup, we recommend that you schedule the signature updates so you always have protection against new vulnerabilities.

- To schedule the signature package download, from configuration mode, specify the start time and the interval for the download:

  ```
  user@host>set security idp security-package automatic interval interval start-time
     <YYYY-MM-DD.HH:MM:SS>
  ```

  For example, to set a schedule for the signature download every 72 hours, you use the following configuration:

  ```
  user@host>set security idp security-package automatic interval 72 start-time
  ```

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the IPS Signature Database on page 306

Verifying the IPS Signature Database

Purpose

Display the IPS signature database.

Action

From operational mode, enter the **show security idp** command.

See Also

- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Understanding the IPS Signature Database on page 298
- Managing the IPS Signature Database (CLI) on page 300

Example: Updating the IPS Signature Database Manually

This example shows how to update the IPS signature database manually.

- Requirements on page 306
- Overview on page 307
- Configuration on page 307
- Verification on page 309

Requirements

Before you begin, configure network interfaces.

Overview

Juniper Networks regularly updates the predefined attack database and makes it available as a security package on the Juniper Networks website. This database includes attack object and attack object groups that you can use in IDP policies to match traffic against known attacks.

In this example, you download the security package with the complete table of attack objects and attack object groups. Once the installation is completed, the attack objects and attack object groups are available in the CLI under the **predefined-attack-groups** and **predefined-attacks** configuration statements at the **[edit security idp idp-policy]** hierarchy level. You create a policy and specify the new policy as the active policy. You only download the updates that Juniper Networks has recently uploaded and then update the attack database, the running policy, and the IPS protocol detector with these new updates.

Configuration

CLI Quick Configuration

CLI quick configuration is not available for this example, because manual intervention is required during the configuration.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To manually download and update the signature database:

1. Specify the URL for the security package.

   ```
   [edit]
   user@host#set security idp security-package url
     https://services.netscreen.com/cgi-bin/index.cgi
   ```

   NOTE: By default it will take URL as https://services.netscreen.com/cgi-bin/index.cgi.

2. Commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

3. Switch to operational mode.

   ```
   [edit]
   user@host# exit
   ```

4. Download the security package.

   ```
   user@host>request security idp security-package download full-update
   ```

5. Check the security package download status.

   ```
   user@host>request security idp security-package download status
   ```

6. Update the attack database using the **install** command.

   ```
   user@host>request security idp security-package install
   ```

7. Check the attack database update status using the following command. The command output displays information about the downloaded and installed versions of attack database versions.

   ```
   user@host>request security idp security-package install status
   ```

8. Switch to configuration mode.

   ```
   user@host>configure
   ```

9. Create an IDP policy.

   ```
   [edit ]
   user@host#edit security idp idp-policy policy1
   ```

10. Associate attack objects or attack object groups with the policy.

    ```
    [edit security idp idp-policy policy1]
    user@host#set rulebase-ips rule rule1 match attacks predefined-attack-groups
      “Response_Critical”
    ```

11. Set action.

    ```
    [edit security idp idp-policy policy1]
    user@host#set rulebase-ips rule rule1 then action no-action
    ```

12. Activate the policy.

    ```
    [edit]
    user@host#set security idp active-policy policy1
    ```

13. Commit the configuration.

    ```
    [edit]
    user@host# commit
    ```

14. In the future if you want to download the signature package, download only the updates that Juniper Networks has recently uploaded.

    ```
    user@host>request security idp security-package download
    ```

15. Check the security package download status.

    ```
    user@host>request security idp security-package download status
    ```

16. Update the attack database, the active policy, and the detector with the new changes.

```
user@host>request security idp security-package install
```

17. Check the attack database, the active policy, and the detector.

```
user@host>request security idp security-package install status
```

NOTE: It is possible that an attack has been removed from the new version of an attack database. If this attack is used in an existing policy on your device, the installation of the new database will fail. An installation status message identifies the attack that is no longer valid. To update the database successfully, remove all references to the deleted attack from your existing policies and groups, and rerun the install command.

Results From configuration mode, confirm your configuration by entering the **show security idp** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show security idp
  idp-policy policy1 {
  rulebase-ips {
    rule rule1 {
  match {
  attacks {
    predefined-attack-groups Response_Critical;
  }
  }
    then {
  action {
  no-action;
  }
}
}
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the IDP Signature Database Manually on page 310

Verifying the IDP Signature Database Manually

Purpose Display the IDP signature database manually.

Action From operational mode, enter the **show security idp** command.

See Also

- Updating the IDP Signature Database Manually Overview on page 34
- Example: Updating the Signature Database Automatically on page 33
- Understanding the IDP Signature Database on page 31

Example: Downloading and Installing the IPS Signature Package in Chassis Cluster Mode

This example shows how to download and install the IPS signature database to a device operating in chassis cluster mode.

- Requirements on page 310
- Overview on page 310
- Downloading and Installing the IPS Signature Database on page 311

Requirements

Before you begin, set the chassis cluster node ID and cluster ID. See *Example: Setting the Node ID and Cluster ID for SRX Series Devices in a Chassis Cluster* .

Overview

The security package for intrusion detection and prevention (IDP) contains a database of predefined IDP attack objects and IDP attack object groups that you can use in IDP policies to match traffic against known and unknown attacks. Juniper Networks regularly updates the predefined attack objects and groups with newly discovered attack patterns.

To update the signature database, you must download a security package from the Juniper Networks website. After downloading the security package, you must install the package to update the security database with the newly downloaded updates from the Staging folder in your device.

NOTE: On branch SRX Series devices, if your device memory utilization is high on the control plane, loading a large IDP policy might cause the device to run out of memory. This can trigger a system reboot during the IPS security package update.

When you download the IPS security package on a device operating in chassis cluster mode, the security package is downloaded to the primary node and then synchronized to the secondary node. This synchronization helps maintain the same version of the security package on both the primary node and the secondary node.

Downloading and Installing the IPS Signature Database

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

1. Specify the URL for the security package.

   ```
   [edit]
   user@host# set security idp security-package url
     https://services.netscreen.com/cgi-bin/index.cgi
   ```

2. Switch to operational mode.

   ```
   [edit]
   user@host# exit
   ```

3. Download the IPS security package to the primary node (downloads in the **var/db/idpd/sec-download** folder).

   ```
   {primary:node0}[edit]
   user@host> request security idp security-package download
   ```

   The following message is displayed:

   ```
   node0:
   --------------------------------------------------------------------------
   Will be processed in async mode. Check the status using the status checking
    CLI
   ```

4. Check the security package download status.

   ```
   {primary:node0}[edit]
   user@host> request security idp security-package download status
   ```

   On a successful download, the following message is displayed.

   ```
   node0:
   --------------------------------------------------------------------------
   Done;Successfully downloaded from
   (https://services.netscreen.com/cgi-bin/index.cgi)
   and synchronized to backup.
   Version info:1871(Mon Mar  7 09:05:30 2011, Detector=11.4.140110223)
   ```

5. Update the attack database using the **install** command.

   ```
   user@host> request security idp security-package install
   ```

6. Check the attack database update status. The command output displays information about the downloaded and installed versions of the attack database.

   ```
   {primary:node0}[edit]
   user@host> request security idp security-package install status

   node0:
   --------------------------------------------------------------------------
   ```

```
Done;Attack DB update : successful - [UpdateNumber=2011,ExportDate=Mon Oct
17 15:13:06 2011,Detector=11.6.140110920]
     Updating control-plane with new detector : successful
     Updating data-plane with new attack or detector : not performed
      due to no existing running policy found.

node1:
--------------------------------------------------------------------------
Done;Attack DB update : successful - [UpdateNumber=2011,ExportDate=Mon Oct
17 15:13:06 2011,Detector=11.6.140110920]
     Updating control-plane with new detector : successful
     Updating data-plane with new attack or detector : not performed
      due to no existing running policy found.
```

NOTE: You must download the IPS signature package to the primary node. This way, the security package is synchronized on the secondary node. Attempts to download the signature package to the secondary node will fail.

If you have configured a scheduled download for the security packages, the signature package files are automatically synchronized from the primary node to the backup node.

See Also

- Understanding Intrusion Prevention System for SRX Series Devices on page 285
- Understanding the IPS Signature Database on page 298
- Managing the IPS Signature Database (CLI) on page 300
- Managing the IPS Signature Database (Security Director) on page 304

Related Documentation

- Introduction to IDP Migration on page 283

CHAPTER 7

Configuration Statements

- from-zone (Security IDP Policy) on page 383
- global (Security IDP) on page 384
- group-members on page 385
- hash-table-size (Security IDP) on page 385
- header-length on page 386
- header-type on page 386
- high-availability (Security IDP) on page 387
- home-address on page 387
- host (Security IDP Sensor Configuration) on page 388
- icmp (Security IDP Custom Attack) on page 388
- icmp (Security IDP Signature Attack) on page 389
- icmpv6 (Security IDP) on page 390
- icmpv6 (Security IDP Custom Attack) on page 391
- identification (Security ICMP Headers) on page 392
- identification (Security IP Headers) on page 393
- idp (Application Services) on page 393
- idp (Security Alarms) on page 394
- idp-policy (Security) on page 395
- ignore-memory-overflow on page 397
- ignore-reassembly-memory-overflow | no-ignore-reassembly-memory-overflow on page 397
- ignore-reassembly-overflow on page 398
- ignore-regular-expression on page 398
- ihl (Security IDP Custom Attack) on page 399
- include-destination-address on page 399
- install on page 400
- interfaces (CoS) on page 401
- interval (Security IDP) on page 402
- ip (Security IDP Custom Attack) on page 402
- ip-action (Security IDP Rulebase IPS) on page 403
- ip-block on page 404
- ip-close on page 404
- ip-connection-rate-limit on page 405
- ip-flags on page 406
- ip-notify on page 406
- ips on page 407
- ipv4 (Security IDP Signature Attack) on page 408

- total-memory on page 496
- to-zone (Security IDP Policy) on page 497
- traceoptions (Security Datapath Debug) on page 498
- traceoptions (Security IDP) on page 500
- ttl (Security IDP) on page 502
- tunable-name on page 503
- tunable-value on page 503
- type (Security IDP Dynamic Attack Group) on page 504
- type (Security IDP ICMP Headers) on page 504
- udp (Security IDP Protocol Binding) on page 505
- udp (Security IDP Signature Attack) on page 506
- udp-anticipated-timeout (Security IDP) on page 506
- urgent-pointer on page 507
- url (Security IDP) on page 507
- weight (Security) on page 508
- window-scale on page 509
- window-size on page 510

ack-number

Supported Platforms SRX Series, vSRX

Syntax

```
ack-number {
  match (equal | greater-than | less-than | not-equal);
  value acknowledgement-number;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the ACK number of the packet. This number identifies the next sequence number; the ACK flag must be set to activate this field.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***acknowledgement-number***—Match the ACK number of the packet.

Range: 0 through 4,294,967,295

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

action (Security Rulebase IPS)

Supported Platforms SRX Series, vSRX

Syntax

```
action {
   class-of-service {
      dscp-code-point number;
      forwarding-class forwarding-class;
   }
   (close-client | close-client-and-server | close-server |drop-connection | drop-packet |
      ignore-connection | mark-diffserv value | no-action | recommended);
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the actions you want IDP to take when the monitored traffic matches the attack objects specified in the rules.

Options

- **no-action**—No action is taken. Use this action when you want to only generate logs for some traffic.
- **ignore-connection**—Stops scanning traffic for the rest of the connection if an attack match is found. IDP disables the rulebase for the specific connection.
- **mark-diffserv** ***value***—Assigns the indicated service-differentiation value to the packet in an attack, then passes them on normally.
- **class-of-service**—Associates a class-of-service forwarding class as an action to the IDP policy; also sets the value of the DSCP code point. You can use the default forwarding class names or define new ones. Forwarding-class and dscp-code-point are optional, but one must be set.
- **drop-packet**—Drops a matching packet before it can reach its destination but does not close the connection. Use this action to drop packets for attacks in traffic that is prone to spoofing, such as UDP traffic. Dropping a connection for such traffic could result in a denial of service that prevents you from receiving traffic from a legitimate source-IP address.
- **drop-connection**—Drops all packets associated with the connection, preventing traffic for the connection from reaching its destination. Use this action to drop connections for traffic that is not prone to spoofing.
- **close-client**—Closes the connection and sends an RST packet to the client but not to the server.
- **close-server**—Closes the connection and sends an RST packet to the server but not to the client.
- **close-client-and-server**—Closes the connection and sends an RST packet to both the client and the server.

- **recommended**—All predefined attack objects have a default action associated with them. This is the action that Juniper Networks recommends when that attack is detected.

NOTE: The actions are listed in the ascending order of severity from low to high. The most severe action is used when there are multiple rule hits for a single session.

Required Privilege Level

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

action-profile

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
action-profile profile-name {
  event (jexec | lbt | lt-enter | lt-leave | mac-egress | mac-ingress | np-egress | np-ingress |
    pot) {
    count;
    packet-dump;
    packet-summary;
    trace;
  }
  module {
    flow {
      flag {
        all;
      }
    }
  }
  preserve-trace-order;
  record-pic-history;
}
```

Hierarchy Level [edit security datapath-debug]

Release Information Command introduced in Junos OS Release 10.0.

Description Configure the action profile options for data path debugging.

Options

- ***action-profile name*** — Name of the action profile.
- **event**—Enable the events to trace the packet when the packet hit the events (jexec, lbt, lt-enter, lt-leave, mac-egress, mac-ingress, np-egress, np-ingress, pot)
  - **count**—Number of times a packet hits the specified event.
  - **packet-dump**—Capture the packet that hits the specified event.
  - **packet-summary**—Print the source/destination IP address details with protocol number and IP length details along with trace message for the specified event.
  - **trace**—Print the standard trace message when the packet hits the specified event.
- **module**—Turn on the flow session related trace messages.
  - **flow**—Trace flow session related messages.
  - **flag**—Specify which flow message needs to be traced.
  - **all**—Trace all possible flow trace messages.
  - **trace**—Print the standard trace message when the packet hits the specified event.
- **preserve-trace-order**—Preserve trace order.

- **record-pic-history**—Record the PICs in which the packet has been processed.

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

Related Documentation

- Example: Configuring Packet Capture for Datapath Debugging on page 277

active-policy

Supported Platforms SRX Series, vSRX

Syntax active-policy *policy-name*;

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify which policy among the configured policies to activate.

Options ***policy-name***—Name of the active policy.

NOTE: You need to make sure the active policy is enforced in the data plane.

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

alert

Supported Platforms SRX Series, vSRX

Syntax alert;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification]

Release Information Statement introduced in Junos OS Release 9.2. .

Description Set an alert flag in the Alert column of the Log Viewer for the matching log record.

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

allow-icmp-without-flow

Supported Platforms SRX Series, vSRX

Syntax

```
(allow-icmp-without-flow | no-allow-icmp-without-flow);
```

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Allow an ICMP packet without matched request. By default the ICMP flow is enabled.

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

anomaly

Supported Platforms SRX Series, vSRX

Syntax

```
anomaly {
   direction (any | client-to-server | server-to-client);
   service service-name;
   shellcode (all | intel | no-shellcode | sparc);
   test test-condition;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type]

Release Information Statement introduced in Junos OS Release 9.3.

Description Protocol anomaly attack objects detect abnormal or ambiguous messages within a connection according to the set of rules for the particular protocol being used.

Options The remaining statements are explained separately. See CLI Explorer.

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

application (Security Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax application *application-name*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the attack for a specified application.

Options *application-name*—Name of the application.

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

application (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax application *application-name*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify an application or an application set name to match.

Options *application-name*—Name of the application.

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

application-identification

Supported Platforms SRX Series, vSRX

Syntax

```
application-identification {
   max-packet-memory-ratio percentage-value;
   max-reass-packet-memory-ratio percentage-value;
   max-tcp-session-packet-memory value;
   max-udp-session-packet-memory value;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2. Packet memory percentages added in Junos OS Release 12.1X44-D20.

Description Enable to identify the TCP/UDP application session running on nonstandard ports to match the application properties of transiting network traffic.

Options define the allocation of IDP memory to application identification for packet and reassembler use.

Options The remaining statements are explained separately. See CLI Explorer.

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

application-services (Security Forwarding Process)

Supported Platforms SRX Series, vSRX

Syntax

```
application-services {
  enable-gtpu-distribution;
  maximize-alg-sessions;
  maximize-idp-sessions {
    weight (firewall | idp);
  }
  packet-ordering-mode {
    (hardware | software);
  }
}
```

Hierarchy Level [edit security forwarding-process]

Release Information Statement introduced in Junos OS Release 9.6. Statement updated in Junos OS Release 10.4. Statement updated in Junos OS Release 15.1X49-D40 with the **enable-gtpu-distribution** option.

Description You can configure SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices to switch from an integrated firewall mode to maximize Intrusion Detection and Prevention (IDP) mode to run IDP processing in tap mode and increase the capacity of processing with the **maximize-idp-sessions** option. Inline tap mode can only be configured if the forwarding process mode is set to **maximize-idp-sessions**, which ensures stability and resiliency for firewall services. You also do not need a separate tap or span port to use inline tap mode. When you maximize IDP, you are decoupling IDP processes from firewall processes, allowing the device to support the same number of firewall and IDP sessions, also run the IDP processing in tap mode..

You can configure maximum Application Layer Gateway (ALG) sessions by using the **maximize-alg-sessions** option. The session capacity number for Real-Time Streaming Protocol (RTSP), FTP, and Trivial File Transfer Protocol (TFTP) ALG varies per flow SPU. For SRX5000 series devices the session capacity is 10,240 per flow SPU. You must reboot the device (and its peer in chassis cluster mode) for the configuration to take effect. The **maximize-alg-sessions** option now enables you to increase defaults as follows:

- TCP proxy connection capacity: 40,000 per flow SPU

NOTE: **Flow session capacity is reduced to half per flow SPU; therefore the aforementioned capacity numbers will not change on central point flow.**

Enable GPRS tunneling protocol. GTP-U session distribution is a UE (User equipment) based distribution, generating tunnel based GTP-U session and distributing them across SPUs on a UE basis.

Before 15.1X49-D40, GTP-U sessions are distributed by GGSN IP address always.

15.1X49-D40 onward, the GTP-U distribution is disabled and fat GTP-U sessions are distributed as normal UDP.

Use the **enable-gtpu-distribution** command to enable GTP-U session distribution.

Options The remaining statements are explained separately. See the CLI Explorer.

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

Related Documentation

- *Juniper Networks Devices Processing Overview*

application-services (Security Policies)

Supported Platforms SRX Series, vSRX

Syntax

```
application-services {
  advanced-anti-malware-policy advanced-anti-malware-policy;
  application-firewall {
    rule-set rule-set;
  }
  application-traffic-control {
    rule-set rule-set;
  }
  gprs-gtp-profile gprs-gtp-profile;
  gprs-sctp-profile gprs-sctp-profile;
  idp idp;
  (redirect-wx redirect-wx | reverse-redirect-wx reverse-redirect-wx);
  security-intelligence-policy security-intelligence-policy;
  ssl-proxy {
    profile-name profile-name;
  }
  uac-policy {
    captive-portal captive-portal;
  }
  utm-policy utm-policy;
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit]

Release Information Statement modified in Junos OS Release 11.1.

Description Enable application services within a security policy. You can enable service such as application firewall, IDP, UTM, SSL proxy, and so on by specifying them in a security policy permit action, when the traffic matches the policy rule.

Options

advanced-anti-malware-policy—Specify advanced-anti-malware policy name.

application-firewall—Specify the rule sets configured as part of application firewall to be applied to the permitted traffic.

application-traffic-control—Specify the rule sets configured as part of AppQoS, application-aware quality of service, to be applied to the permitted traffic.

gprs-gtp-profile—Specify GPRS tunneling protocol profile name.

gprs-sctp-profile—Specify GPRS stream control protocol profile name.

idp—Apply Intrusion detection and prevention (IDP) as application services.

redirect-wx—Specify the WX redirection needed for the packets that arrive from the LAN.

reverse-redirect-wx—Specify the WX redirection needed for the reverse flow of the packets that arrive from the WAN.

security-intelligence-policy—Specify security-intelligence policy name.

uac-policy —Enable Unified Access Control (UAC) for the security policy. This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a UAC deployment.

captive-portal ***captive-portal***—Specify the preconfigured security policy for captive portal on the Junos OS Enforcer to enable the captive portal feature. The captive portal policy is configured as part of the UAC policy. By configuring the captive portal feature, you can redirect traffic destined for protected resources to the IC Series device or to the URL you configure on the Junos OS Enforcer.

utm-policy ***utm-policy***—Specify UTM policy name. The UTM policy configured for antivirus, antispam, content-filtering, traffic-options, and Web-filtering protocols is attached to the security policy to be applied to the permitted traffic.

Required Privilege Level

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

Related Documentation

- *Application Firewall Overview*

attack-type (Security Anomaly)

Supported Platforms

SRX Series, vSRX

Syntax

```
attack-type {
   anomaly {
      direction (any | client-to-server | server-to-client);
      service service-name;
      shellcode (all | intel | no-shellcode | sparc);
      test test-condition;
   }
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name*]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Specify the type of attack.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

attack-type (Security Chain)

Supported Platforms SRX Series, vSRX

Syntax

```
attack-type {
  chain {
    expression boolean-expression;
    member member-name {
      attack-type {
        (anomaly ...same statements as in [edit security idp custom-attack attack-name
          attack-type anomaly] hierarchy level | signature ...same statements as in [edit
          security idp custom-attack attack-name attack-type signature] hierarchy level);
      }
    }
    order;
    protocol-binding {
      application application-name;
      icmp;
      icmpv6;
      ip {
        protocol-number transport-layer-protocol-number;
      }
      ipv6 {
        protocol-number transport-layer-protocol-number;
      }
      rpc {
        program-number rpc-program-number;
      }
      tcp {
        minimum-port port-number <maximum-port port-number>;
      }
      udp {
        minimum-port port-number <maximum-port port-number>;
      }
    }
    reset;
    scope (session | transaction);
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the type of attack.

NOTE: In a chain attack, you can configure multiple member attacks.

In an attack, under protocol binding TCP/UDP, you can specify multiple ranges of ports.

Options The remaining statements are explained separately. See CLI Explorer.

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

attack-type (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
attack-type {
anomaly {
  direction (any | client-to-server | server-to-client);
  shellcode (all | intel |no-shellcode | sparc);
  test-condition condition-name;
}
signature {
  context context-name;
  direction (any | client-to-server | server-to-client);
  negate;
  pattern signature-pattern;
  pattern-pcre signature-pattern-pcre;
  protocol {
    icmp {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      code {
        match (equal | greater-than | less-than | not-equal);
        value code-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value data-length;
      }
      identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
      }
      sequence-number {
        match (equal | greater-than | less-than | not-equal);
        value sequence-number;
      }
      type {
        match (equal | greater-than | less-than | not-equal);
        value type-value;
      }
    }
    icmpv6 {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      code {
        match (equal | greater-than | less-than | not-equal);
        value code-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value data-length;
      }
      identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
      }
      sequence-number {
        match (equal | greater-than | less-than | not-equal);
        value sequence-number;
      }
      type {
        match (equal | greater-than | less-than | not-equal);
        value type-value;
      }
    }
    ipv4 {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
      }
      identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
      }
      ip-flags {
        (df | no-df);
        (mf | no-mf);
        (rb | no-rb);
      }
      protocol {
        match (equal | greater-than | less-than | not-equal);
        value transport-layer-protocol-id;
      }
      source {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
      }
      tos {
        match (equal | greater-than | less-than | not-equal);
        value type-of-service-in-decimal;
      }
      total-length {
        match (equal | greater-than | less-than | not-equal);
        value total-length-of-ip-datagram;
      }
      ttl {
        match (equal | greater-than | less-than | not-equal);
        value time-to-live;
      }
    }
    ipv6 {
      destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
      }
      extension-header {
        destination-option {
          home-address {
            match (equal | greater-than | less-than | not-equal);
            value header-value;
          }
            option-type {
              match (equal | greater-than | less-than | not-equal);
              value header-value;
            }
        }
        routing-header {
          header-type {
            match (equal | greater-than | less-than | not-equal);
            value header-value;
            }
          }
      }
      flow-label {
        match (equal | greater-than | less-than | not-equal);
        value flow-label-value;
      }
      hop-limit {
        match (equal | greater-than | less-than | not-equal);
        value hop-limit-value;
      }
      next-header {
        match (equal | greater-than | less-than | not-equal);
        value next-header-value;
      }
      payload-length {
        match (equal | greater-than | less-than | not-equal);
        value payload-length-value;
      }
      source {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
      }
      traffic-class {
        match (equal | greater-than | less-than | not-equal);
        value traffic-class-value;
      }
    tcp {
      ack-number {
        match (equal | greater-than | less-than | not-equal);
        value acknowledgement-number;
      }
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value tcp-data-length;
      }
      destination-port {
        match (equal | greater-than | less-than | not-equal);
        value destination-port;
      }
      header-length {
        match (equal | greater-than | less-than | not-equal);
        value header-length;
      }
      mss {
        match (equal | greater-than | less-than | not-equal);
        value maximum-segment-size;
      }
      option {
        match (equal | greater-than | less-than | not-equal);
        value tcp-option;
      }
      reserved {
        match (equal | greater-than | less-than | not-equal);
        value reserved-value;
      }
      sequence-number {
        match (equal | greater-than | less-than | not-equal);
        value sequence-number;
      }
      source-port {
        match (equal | greater-than | less-than | not-equal);
        value source-port;
      }
      tcp-flags {
        (ack | no-ack);
        (fin | no-fin);
        (psh | no-psh);
        (r1 | no-r1);
        (r2 | no-r2);
        (rst | no-rst);
        (syn | no-syn);
        (urg | no-urg);
      }
      urgent-pointer {
        match (equal | greater-than | less-than | not-equal);
        value urgent-pointer;
      }
      window-scale {
        match (equal | greater-than | less-than | not-equal);
        value window-scale-factor;
      }
      window-size {
        match (equal | greater-than | less-than | not-equal);
        value window-size;
      }
    }
    udp {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
```

```
        value checksum-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value data-length;
      }
      destination-port {
        match (equal | greater-than | less-than | not-equal);
        value destination-port;
      }
      source-port {
        match (equal | greater-than | less-than | not-equal);
        value source-port;
      }
    }
  }
  protocol-binding {
    application application-name;
    icmp;
    icmpv6;
    ip {
      protocol-number transport-layer-protocol-number;
    }
    ipv6 {
      protocol-number transport-layer-protocol-number;
    }
    rpc {
      program-number rpc-program-number;
    }
    tcp {
      minimum-port port-number <maximum-port port-number>;
    }
    udp {
      minimum-port port-number <maximum-port port-number>;
    }
  }
  regexp regular-expression;
  shellcode (all | intel | no-shellcode | sparc);
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain member *member-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the type of attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

attack-type (Security Signature)

Supported Platforms SRX Series, vSRX

Syntax

```
attack-type {
  signature {
    context context-name;
    direction (any | client-to-server | server-to-client);
    negate;
    pattern signature-pattern;
    pattern-pcre signature-pattern-pcre;
    protocol {
      icmp {
        code {
          match (equal | greater-than | less-than | not-equal);
          value code-value;
        }
        data-length {
          match (equal | greater-than | less-than | not-equal);
          value data-length;
        }
        identification {
          match (equal | greater-than | less-than | not-equal);
          value identification-value;
        }
        sequence-number {
          match (equal | greater-than | less-than | not-equal);
          value sequence-number;
        }
        type {
          match (equal | greater-than | less-than | not-equal);
          value type-value;
        }
      }
      icmpv6 {
        code {
          match (equal | greater-than | less-than | not-equal);
          value code-value;
        }
        data-length {
          match (equal | greater-than | less-than | not-equal);
          value data-length;
        }
        identification {
          match (equal | greater-than | less-than | not-equal);
          value identification-value;
        }
        sequence-number {
          match (equal | greater-than | less-than | not-equal);
          value sequence-number;
        }
        type {
          match (equal | greater-than | less-than | not-equal);
          value type-value;
    }
}
ipv4 {
    destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
    }
    ihl {
        match (equal | greater-than | less-than | not-equal);
        value ihl-value;
    }
    ip-flags {
        (df | no-df);
        (mf | no-mf);
        (rb | no-rb);
    }
    protocol {
        match (equal | greater-than | less-than | not-equal);
        value transport-layer-protocol-id;
    }
    source {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    tos {
        match (equal | greater-than | less-than | not-equal);
        value type-of-service-in-decimal;
    }
    total-length {
        match (equal | greater-than | less-than | not-equal);
        value total-length-of-ip-datagram;
    }
    ttl {
        match (equal | greater-than | less-than | not-equal);
        value time-to-live;
    }
}
ipv6 {
    destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    flow-label {
        match (equal | greater-than | less-than | not-equal);
        value flow-label-value;
    }
    hop-limit {
        match (equal | greater-than | less-than | not-equal);
        value hop-limit-value;
    }
    next-header {
        match (equal | greater-than | less-than | not-equal);
```

```
            value next-header-value;
        }
        payload-length {
            match (equal | greater-than | less-than | not-equal);
            value payload-length-value;
        }
        source {
            match (equal | greater-than | less-than | not-equal);
            value ip-address-or-hostname;
        }
        traffic-class {
            match (equal | greater-than | less-than | not-equal);
            value traffic-class-value;
        }
    tcp {
        ack-number {
            match (equal | greater-than | less-than | not-equal);
            value acknowledgement-number;
        }
        data-length {
            match (equal | greater-than | less-than | not-equal);
            value tcp-data-length;
        }
        destination-port {
            match (equal | greater-than | less-than | not-equal);
            value destination-port;
        }
        header-length {
            match (equal | greater-than | less-than | not-equal);
            value header-length;
        }
        mss {
            match (equal | greater-than | less-than | not-equal);
            value maximum-segment-size;
        }
        option {
            match (equal | greater-than | less-than | not-equal);
            value tcp-option;
        }
        sequence-number {
            match (equal | greater-than | less-than | not-equal);
            value sequence-number;
        }
        source-port {
            match (equal | greater-than | less-than | not-equal);
            value source-port;
        }
        tcp-flags {
            (ack | no-ack);
            (fin | no-fin);
            (psh | no-psh);
            (r1 | no-r1);
            (r2 | no-r2);
            (rst | no-rst);
            (syn | no-syn);
            (urg | no-urg);
        }
        urgent-pointer {
          match (equal | greater-than | less-than | not-equal);
          value urgent-pointer;
        }
        window-scale {
          match (equal | greater-than | less-than | not-equal);
          value window-scale-factor;
        }
        window-size {
          match (equal | greater-than | less-than | not-equal);
          value window-size;
        }
      }
      udp {
        data-length {
          match (equal | greater-than | less-than | not-equal);
          value data-length;
        }
        destination-port {
          match (equal | greater-than | less-than | not-equal);
          value destination-port;
        }
        source-port {
          match (equal | greater-than | less-than | not-equal);
          value source-port;
        }
      }
    }
    protocol-binding {
      application application-name;
      icmp;
      icmpv6;
      ip {
        protocol-number transport-layer-protocol-number;
      }
      ipv6 {
        protocol-number transport-layer-protocol-number;
      }
      rpc {
        program-number rpc-program-number;
      }
      tcp {
        minimum-port port-number <maximum-port port-number>;
      }
      udp {
        minimum-port port-number <maximum-port port-number>;
      }
    }
    regexp regular-expression;
    shellcode (all | intel | no-shellcode | sparc);
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the type of attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

attacks (Security Exempt Rulebase)

Supported Platforms SRX Series, vSRX

Syntax

```
attacks {
   custom-attack-groups [attack-group-name];
   custom-attacks [attack-name];
   dynamic-attack-groups [attack-group-name];
   predefined-attack-groups [attack-group-name];
   predefined-attacks [attack-name];
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the attacks that you do not want the device to match in the monitored network traffic. Each attack is defined as an attack object, which represents a known pattern of attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

attacks (Security IPS Rulebase)

Supported Platforms SRX Series, vSRX

Syntax

```
attacks {
   custom-attack-groups [attack-group-name];
   custom-attacks [attack-name];
   dynamic-attack-groups [attack-group-name];
   predefined-attack-groups [attack-group-name];
   predefined-attacks [attack-name];
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the attacks you want the device to match in the monitored network traffic. Each attack is defined as an attack object, which represents a known pattern of attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

automatic (Security)

Supported Platforms SRX Series, vSRX

Syntax

```
automatic {
   download-timeout minutes;
   enable;
   interval hours;
   start-time start-time;
}
```

Hierarchy Level [edit security idp security-package]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable the device to automatically download the updated signature database from the specified URL.

Options The remaining statements are explained separately. See CLI Explorer.

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

cache-prune-chunk-size

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax cache-prune-chunk-size *number*;

Hierarchy Level [edit security idp sensor-configuration ssl-inspection]

Release Information Statement introduced in Junos OS Release 10.2.

Description Number of cache entries to delete when pruning SSL session ID cache.

Options ***cache-prune-chunk-size***—Number of cache entries to delete when pruning SSL session ID cache.

Range: 1 through 100,000

Default: 10,000

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

cache-size (Security)

Supported Platforms SRX Series, vSRX

Syntax cache-size *size*;

Hierarchy Level [edit security idp sensor-configuration log]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the size in bytes for each user’s log cache.

Options **size**—Cache size.

Range: 1 through 65,535 bytes

Default: 12,800 bytes

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

category (Security Dynamic Attack Group)

Supported Platforms SRX Series, vSRX

Syntax

```
category {
  values [category-value];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a category filter to add attack objects based on the category.

Options **values**—Name of the category filter. You can configure multiple filters separated by spaces and enclosed in square brackets.

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

chain

Supported Platforms SRX Series, vSRX

Syntax

```
chain {
  expression boolean-expression;
  member member-name {
    attack-type {
      (anomaly ...same statements as in [edit security idp custom-attack attack-name
        attack-type anomaly] hierarchy level | signature ...same statements as in [edit security
        idp custom-attack attack-name attack-type signature] hierarchy level);
    }
  }
  order;
  protocol-binding {
    application application-name;
    icmp;
    icmpv6;
    ip {
      protocol-number transport-layer-protocol-number;
    }
    ipv6 {
      protocol-number transport-layer-protocol-number;
    }
    rpc {
      program-number rpc-program-number;
    }
    tcp {
      minimum-port port-number <maximum-port port-number>;
    }
    udp {
      minimum-port port-number <maximum-port port-number>;
    }
  }
  reset;
  scope (session | transaction);
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type]

Release Information Statement introduced in Junos OS Release 9.3.

Description Chain attack object combines multiple signatures and/or protocol anomalies into a single object. Traffic must match all of the combined signatures and/or protocol anomalies to match the chain attack object.

Options The remaining statements are explained separately. See CLI Explorer.

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

checksum-validate

Supported Platforms SRX Series

Syntax

```
checksum-validate {
   match (equal | greater-than | less-than | not-equal);
   value checksum-value;
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]
[edit security idp custom-attack *attack-name* attack-type signature protocol tcp]
[edit security idp custom-attack *attack-name* attack-type signature protocol udp]
[edit security idp custom-attack *attack-name* attack-type signature protocol icmp]
[edit security idp custom-attack *attack-name* attack-type signature protocol icmpv6]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Allow IDP to validate checksum field against the calculated checksum.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value *checksum-value*—Match a decimal value.

Range: 0 through 65,535

Required Privilege Level

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

classifiers (CoS)

Supported Platforms SRX Series, vSRX

Syntax

```
classifiers {
  (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) classifier-name {
    forwarding-class forwarding-class-name {
    loss-priority (high | low | medium-high | medium-low) {
      code-point alias-or-bit-string ;
    }
    import (default | user-defined;
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 9.2

Description Configure a user-defined behavior aggregate (BA) classifier.

Options

- *classifier-name*—User-defined name for the classifier.
- import (default | *user-defined*)—Specify the template to use to map any code points not explicitly mapped in this configuration. For example, if the classifier is of type **dscp** and you specify **import default**, code points you do not map in your configuration will use the predefined DSCP default mapping; if you specify **import mymap**, for example, code points not mapped in the forwarding-class configuration would use the mappings in a user-defined classifier named **mymap**.
- forwarding-class *class-name*—Specify the name of the forwarding class. You can use the default forwarding class names or define new ones.
- loss-priority *level*—Specify a loss priority for this forwarding class: **high**, **low**, **medium-high**, **medium-low**.
- code-points (*alias* | *bits*)—Specify a code-point alias or the code points that map to this forwarding class.

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

Related Documentation

- *Understanding Interfaces*

code

Supported Platforms SRX Series, vSRX

Syntax

```
code {
  match (equal | greater-than | less-than | not-equal);
  value code-value;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol icmp]
[edit security idp custom-attack *attack-name* attack-type signature protocol icmpv6]

Release Information Statement introduced in Junos OS Release 9.3. Statement modified in Junos OS Release 12.3X48-D25 to add ICMPv6 protocol support for custom attacks.

Description Specify the secondary code that identifies the function of the request/reply within a given type.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***code-value***—Match a decimal value.

Range: 0 through 255

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

code-points (CoS)

Supported Platforms SRX Series, vSRX

Syntax code-points [*aliases*] [*6–bit-patterns*];

Hierarchy Level [edit class-of-service classifiers *type classifier-name* forwarding-class *class-name*]

Release Information Statement introduced in Junos OS Release 8.5.

Description Specify one or more DSCP code-point aliases or bit sets for association with a forwarding class.

Options *aliases*—Name of the DSCP alias.

6–bit patterns—Value of the code-point bits, in decimal form.

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

context (Security Custom Attack)

Supported Platforms	SRX Series, vSRX
Syntax	context *context-name*;
Hierarchy Level	[edit security idp custom-attack *attack-name* attack-type signature]
Release Information	Statement introduced in Junos OS Release 9.3.
Description	Define the location of the signature where IDP should look for the attack in a specific Application Layer protocol.
Options	***context-name***—Name of the context under which the attack has to be matched.
Required Privilege Level	security—To view this statement in the configuration. security-control—To add this statement to the configuration.

content-decompression-max-memory-kb

Supported Platforms SRX Series, vSRX

Syntax content-decompression-max-memory-kb *value*;

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 11.2.

Description Set the maximum memory allocation in kilobytes for content decompression.

The default memory allocation provides 33 KB per session for an average number of sessions requiring decompression at the same time. To determine if this value is consistent with your environment, analyze values from decompression-related counters and the total number of IDP sessions traversing the device. Estimate the number of sessions requiring decompression at the same time. Assuming that each of these sessions requires 33 KB of memory for decompression, compare your estimated needs to the default value.

NOTE: Because content decompression requires a significant allocation of memory, system performance will be impacted by increasing the maximum memory allocation for decompression.

Options **Range:** 50 through 2,000,000 KB

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

content-decompression-max-ratio

Supported Platforms SRX Series, vSRX

Syntax content-decompression-max-ratio *value*;

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 11.2.

Description Set the maximum decompression ratio of the size of decompressed data to the size of compressed data.

Some attacks are introduced through compressed content. When the content is decompressed, it can inflate to a very large size taking up valuable system resources resulting in denial of service. This type of attack can be recognized by the ratio of the size of decompressed data to the size of compressed data. Keep in mind, however, that a higher ratio lessens the chance of detecting this type of attack.

Options **Range:** 1 through 128

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

count (Security Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax count *count-value*;

Hierarchy Level [edit security idp custom-attack *attack-name* time-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the number of times that IDP detects the attack within the specified scope before triggering an event.

Options ***count-value***—Number of times IDP detects the attack.

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

custom-attack

Supported Platforms SRX Series, vSRX

Syntax

```
custom-attack attack-name {
  attack-type {
    anomaly {
      direction (any | client-to-server | server-to-client);
      service service-name;
      shellcode (all | intel | no-shellcode | sparc);
      test test-condition;
    }
    chain {
      expression boolean-expression;
      member member-name {
        attack-type {
          (anomaly ...same statements as in [edit security idp custom-attack attack-name
            attack-type anomaly] hierarchy level | signature ...same statements as in [edit
            security idp custom-attack attack-name attack-type signature] hierarchy level);
        }
      }
      order;
      protocol-binding {
        application application-name;
        icmp;
        icmpv6;
        ip {
          protocol-number transport-layer-protocol-number;
        }
        ipv6 {
          protocol-number transport-layer-protocol-number;
        }
        rpc {
          program-number rpc-program-number;
        }
        tcp {
          minimum-port port-number <maximum-port port-number>;
        }
        udp {
          minimum-port port-number <maximum-port port-number>;
        }
      }
      reset;
      scope (session | transaction);
    }
    signature {
      context context-name;
      direction (any | client-to-server | server-to-client);
      negate;
      pattern signature-pattern;
      pattern-pcre signature-pattern-pcre;
      protocol {
        icmp {
          checksum-validate {
```

```
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  code {
    match (equal | greater-than | less-than | not-equal);
    value code-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value data-length;
  }
  identification {
    match (equal | greater-than | less-than | not-equal);
    value identification-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  type {
    match (equal | greater-than | less-than | not-equal);
    value type-value;
  }
}
icmpv6 {
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  code {
    match (equal | greater-than | less-than | not-equal);
    value code-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value data-length;
  }
  identification {
    match (equal | greater-than | less-than | not-equal);
    value identification-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  type {
    match (equal | greater-than | less-than | not-equal);
    value type-value;
  }
}
ipv4 {
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
    }
    ihl {
        match (equal | greater-than | less-than | not-equal);
        value ihl-value;
    }
    ip-flags {
        (df | no-df);
        (mf | no-mf);
        (rb | no-rb);
    }
    protocol {
        match (equal | greater-than | less-than | not-equal);
        value transport-layer-protocol-id;
    }
    source {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    tos {
        match (equal | greater-than | less-than | not-equal);
        value type-of-service-in-decimal;
    }
    total-length {
        match (equal | greater-than | less-than | not-equal);
        value total-length-of-ip-datagram;
    }
    ttl {
        match (equal | greater-than | less-than | not-equal);
        value time-to-live;
    }
}
ipv6 {
    destination {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    flow-label {
        match (equal | greater-than | less-than | not-equal);
        value flow-label-value;
    }
    hop-limit {
        match (equal | greater-than | less-than | not-equal);
        value hop-limit-value;
    }
    next-header {
        match (equal | greater-than | less-than | not-equal);
        value next-header-value;
    }
    payload-length {
        match (equal | greater-than | less-than | not-equal);
    value payload-length-value;
  }
  source {
    match (equal | greater-than | less-than | not-equal);
    value ip-address-or-hostname;
  }
  traffic-class {
    match (equal | greater-than | less-than | not-equal);
    value traffic-class-value;
  }
tcp {
  ack-number {
    match (equal | greater-than | less-than | not-equal);
    value acknowledgement-number;
  }
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value tcp-data-length;
  }
  destination-port {
    match (equal | greater-than | less-than | not-equal);
    value destination-port;
  }
  header-length {
    match (equal | greater-than | less-than | not-equal);
    value header-length;
  }
  mss {
    match (equal | greater-than | less-than | not-equal);
    value maximum-segment-size;
  }
  option {
    match (equal | greater-than | less-than | not-equal);
    value tcp-option;
  }
  reserved {
    match (equal | greater-than | less-than | not-equal);
    value reserved-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  source-port {
    match (equal | greater-than | less-than | not-equal);
    value source-port;
  }
  tcp-flags {
    (ack | no-ack);
    (fin | no-fin);
    (psh | no-psh);
    (r1 | no-r1);
```

```
         (r2 | no-r2);
         (rst | no-rst);
         (syn | no-syn);
         (urg | no-urg);
      }
      urgent-pointer {
         match (equal | greater-than | less-than | not-equal);
         value urgent-pointer;
      }
      window-scale {
         match (equal | greater-than | less-than | not-equal);
         value window-scale-factor;
      }
      window-size {
         match (equal | greater-than | less-than | not-equal);
         value window-size;
      }
   }
   udp {
      checksum-validate {
         match (equal | greater-than | less-than | not-equal);
         value checksum-value;
      }
      data-length {
         match (equal | greater-than | less-than | not-equal);
         value data-length;
      }
      destination-port {
         match (equal | greater-than | less-than | not-equal);
         value destination-port;
      }
      source-port {
         match (equal | greater-than | less-than | not-equal);
         value source-port;
      }
   }
}
protocol-binding {
   application application-name;
   icmp;
   icmpv6;
   ip {
      protocol-number transport-layer-protocol-number;
   }
   ipv6 {
      protocol-number transport-layer-protocol-number;
   }
   rpc {
      program-number rpc-program-number;
   }
   tcp {
      minimum-port port-number <maximum-port port-number>;
   }
   udp {
      minimum-port port-number <maximum-port port-number>;
   }
```

```
                }
                regexp regular-expression;
                shellcode (all | intel | no-shellcode | sparc);
            }
        }
        recommended-action (close | close-client | close-server | drop | drop-packet | ignore |
            none);
        severity (critical | info | major | minor | warning);
        time-binding {
            count count-value;
            scope (destination | peer | source);
        }
    }
```

Hierarchy Level [edit security idp]

Release Information Statement modified in Junos OS Release 9.3.

Description Configure custom attack objects to detect a known or unknown attack that can be used to compromise your network.

Options ***attack-name***—Name of the custom attack object. The maximum number of characters allowed for a custom attack object name is 60.

The remaining statements are explained separately. See CLI Explorer.

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

custom-attack-group

Supported Platforms SRX Series, vSRX

Syntax

```
custom-attack-group custom-attack-group-name {
  group-members [attack-or-attack-group-name];
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Configure custom attack group. A custom attack group is a list of attacks that would be matched on the traffic if the group is selected in a policy.

Options *custom-attack-group-name*—Name of the custom attack group.

The remaining statements are explained separately. See CLI Explorer.

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

custom-attack-groups (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
custom-attack-groups attack-group-name;
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match attacks]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match attacks]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a name for the custom attack group.

Options *attack-group-name*—Name of the custom attack group.

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

custom-attacks

Supported Platforms SRX Series, vSRX

Syntax

```
custom-attacks [attack-name];
```

Hierarchy Level

```
[edit security idp idp-policy policy-name rulebase-exempt rule rule-name match attacks],
[edit security idp idp-policy policy-name rulebase-ips rule rule-name match attacks]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Select custom attacks defined under [**edit security idp custom-attack**] by specifying their names.

Options ***attack-name***—Name of the new custom attack object.

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

data-length

Supported Platforms SRX Series, vSRX

Syntax

```
data-length {
  match (equal | greater-than | less-than | not-equal);
  value tcp-data-length;
}
```

Hierarchy Level

```
[edit security idp custom-attack attack-name attack-type signature protocol udp]
[edit security idp custom-attack attack-name attack-type signature protocol icmp]
[edit security idp custom-attack attack-name attack-type signature protocol icmpv6]
[edit security idp custom-attack attack-name attack-type signature protocol tcp]
```

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the number of bytes in the data payload. In the TCP header, for SYN, ACK, and FIN packets, this field should be empty.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***data-length***—Match the number of bytes in the data payload.

Range: 0 through 65,535

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

datapath-debug

Supported Platforms SRX Series

Syntax

```
datapath-debug {
  action-profile profile-name {
    event (jexec | lbt | lt-enter | lt-leave | mac-egress | mac-ingress | np-egress | np-ingress
      | pot) {
      count;
      packet-dump;
      packet-summary;
      trace;
    }
    module {
      flow {
        flag {
          all;
        }
      }
    }
    preserve-trace-order;
    record-pic-history;
  }
  capture-file {
    filename;
    files number;
    format pacp-format;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  maximum-capture-size value;
  packet-filter packet-filter-name {
    action-profile (profile-name | default);
    destination-port (port-range | protocol-name);
    destination-prefix destination-prefix;
    interface logical-interface-name;
    protocol (protocol-number | protocol-name;
    source-port (port-range | protocol- name);
    source-prefix source-prefix;
    }
  traceoptions {
    file {
      filename;
      files number;
      match regular-expression;
      size maximum-file-size;
      (world-readable | no-world-readable);
    }
    no-remote-trace;
  }
}
```

Hierarchy Level [edit security]

Release Information Command introduced in Junos OS Release 10.0.

Description Configure the data path debugging options.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Understanding Data Path Debugging for Logical Systems*

description (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax description *text*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name*]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name*]

Release Information Statement modified in Junos OS Release 9.2.

Description Specify descriptive text for an exempt rule, or IPS rule.

Options *text*—Descriptive text about an exempt rule, or IPS rule.

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

destination (Security IP Headers Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
destination {
  match (equal | greater-than | less-than | not-equal);
  value ip-address-or-hostname;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]
[edit security idp custom-attack *attack-name* attack-type signature protocol ipv6]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the IP address of the attack target.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value *ip-address-or-hostname***—Match an IP address or a hostname.

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

destination-address (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax

```
destination-address ([address-name] | any | any-ipv4 | any-ipv6);
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a destination IP address or IP address set object to be used as the match destination address object. The default value is any.

Options

- ***address-name***—IP address or IP address set object.
- ***any***—Specify any IPv4 or IPv6 address.
- ***any-ipv4***—Specify any IPv4 address.
- ***any-ipv6***—Specify any IPv6 address.

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

destination-except

Supported Platforms SRX Series, vSRX

Syntax

```
destination-except [address-name];
```

Hierarchy Level

```
[edit security idp idp-policy policy-name rulebase-exempt rule rule-name match]
[edit security idp idp-policy policy-name rulebase-ips rule rule-name match]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a destination IP address or IP address set object to specify all destination address objects except the specified address objects. The default value is any.

Options ***address-name***—IP address or IP address set object.

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

destination-option

Supported Platforms SRX Series

Syntax

```
destination-option {
  home-address {
    match (equal | greater-than | less-than | not-equal);
    value header-value;
  }
  option-type {
    match (equal | greater-than | less-than | not-equal);
    value header-value;
  }
}
```

Hierarchy Level

```
[edit set security idp custom-attack attack-name attack-type signature protocol ipv6
  extension-header]
```

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv6 destination option for the extension header. The **destination-option** option inspects the header option type of **home-address** field in the **extension header** and reports a custom attack if a match is found. The **destination-option** supports the **home-address** field type of inspection.

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

destination-port (Security Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
destination-port {
  match (equal | greater-than | less-than | not-equal);
  value destination-port;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol udp]
[edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the port number of the attack target.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***destination-port***—Match the port number of the attack target.

Range: 0 through 65,535

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

detect-shellcode

Supported Platforms SRX Series, vSRX

Syntax

```
(detect-shellcode | no-detect-shellcode);
```

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable to detect the shell code and prevent buffer overflow attacks. By default this setting is enabled.

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

detector

Supported Platforms SRX Series, vSRX

Syntax

```
detector {
  protocol-name protocol-name {
    tunable-name  tunable-name {
      tunable-value protocol-value;
    }
  }
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure protocol detector engine for a specific service.

Options The remaining statements are explained separately. See CLI Explorer.

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

direction (Security Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax direction (any | client-to-server | server-to-client);

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type anomaly]
[edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Define the connection direction of the attack.

Options

- **any**—Detect the attack in either direction.
- **client-to-server**—Detect the attack only in client-to-server traffic.
- **server-to-client**—Detect the attack only in server-to-client traffic.

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

direction (Security Dynamic Attack Group)

Supported Platforms SRX Series, vSRX

Syntax

```
direction {
  expression (and | or);
  values [any client-to-server exclude-any exclude-client-to-server exclude-server-to-client
    server-to-client];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3. The **expression** option added in Junos OS Release 11.4.

Description Specify a direction filter to add predefined attacks to the dynamic group based on the direction specified in the attacks.

Options **expression**—Boolean operators:

- **and**— If both the member name patterns match, the expression matches.
- **or**— If either of the member name patterns match, the expression matches.

values—Name of the direction filter. You can select from the following directions:

- **any**—Monitors traffic from client to server and server to client.
- **client-to-server**—Monitors traffic from client to server (most attacks occur over **client-to-server** connections) only.
- **exclude-any**—Allows traffic from client to server and server to client.
- **exclude-client-to-server**—Allows traffic from client to server only.
- **exclude-server-to-client**—Allows traffic from server to client only.
- **server-to-client**—Monitors traffic from server to client only.

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

download-timeout

Syntax download-timeout *minutes*;

Hierarchy Level [edit security idp security-package automatic]

Release Information Statement introduced in Release 9.6 R3 of Junos OS.

Description Specify the time that the device automatically times out and stops downloading the updated signature database from the specified URL.

NOTE: The default value for download-timeout is one minute. If download is completed before the download times out, the signature is automatically updated after the download. If the download takes longer than the configured period, the automatic signature update is aborted.

Options *minutes*—Time in minutes.

Range: 1 through 60 minutes

Default: 1 minute

NOTE: For SRX Series devices the applicable range is 1 through 4000000 per second.

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

drop-if-no-policy-loaded

Supported Platforms SRX Series, vSRX

Syntax drop-if-no-policy-loaded;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description Drop all traffic until the IDP policy gets loaded.

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

drop-on-failover

Supported Platforms	SRX Series, vSRX
Syntax	drop-on-failover;
Hierarchy Level	[edit security idp sensor-configuration flow]
Release Information	Statement introduced in Junos OS Release 12.1X44-D20.
Description	Drop traffic on chassis cluster failover sessions.
Required Privilege Level	security—To view this statement in the configuration. security-control—To add this statement to the configuration.

drop-on-limit

Supported Platforms	SRX Series, vSRX
Syntax	drop-on-limit;
Hierarchy Level	[edit security idp sensor-configuration flow]
Release Information	Statement introduced in Junos OS Release 12.1X44-D20.
Description	Drop connections on exceeding resource limits.
Required Privilege Level	security—To view this statement in the configuration. security-control—To add this statement to the configuration.

dynamic-attack-group

Supported Platforms SRX Series, vSRX

Syntax

```
dynamic-attack-group dynamic-attack-group-name {
  filters {
    category {
      values [category-value];
    }
    direction {
      expression (and | or);
      values [any client-to-server exclude-any exclude-client-to-server
        exclude-server-to-client server-to-client];
    }
    false-positives {
      values [frequently occasionally rarely unknown];
    }
    performance {
      values [fast normal slow unknown];
    }
    products {
      values [product-value];
    }
    recommended;
    service {
      values [service-value];
    }
    severity {
      values [critical info major minor warning];
    }
    type {
      values [anomaly signature];
    }
  }
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.3. The **expression** option added in Junos OS Release 11.4.

Description Configure a dynamic attack group. A dynamic attack group selects its members based on the filters specified in the group. Therefore, the list of attacks is updated (added or removed) when a new signature database is used.

Options ***dynamic-attack-group-name***—Name of the dynamic attack group.

The remaining statements are explained separately. See CLI Explorer.

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

dynamic-attack-groups (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
dynamic-attack-groups attack-group-name;
```

Hierarchy Level

```
[edit security idp idp-policy policy-name rulebase-exempt rule rule-name match attacks]
[edit security idp idp-policy policy-name rulebase-ips rule rule-name match attacks]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a name for the dynamic attack group.

Options *attack-group-name*—Name of the dynamic attack group.

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

enable

Supported Platforms SRX Series, vSRX

Syntax

```
enable {
   download-timeout minutes;
   interval hours;
   start-time start-time;
}
```

Hierarchy Level

```
[edit security idp security-package automatic]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Enables the automatic download of the IDP security package.

Options The remaining statements are explained separately. See CLI Explorer.

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

enable-all-qmodules

Supported Platforms SRX Series, vSRX

Syntax (enable-all-qmodules | no-enable-all-qmodules);

Hierarchy Level [edit security idp sensor-configuration global]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable all the qmodules of the global rulebase IDP security policy. By default all the qmodules are enabled.

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

enable-packet-pool

Supported Platforms SRX Series, vSRX

Syntax (enable-packet-pool | no-enable-packet-pool);

Hierarchy Level [edit security idp sensor-configuration global]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable the packet pool to use when the current pool is exhausted. By default packet pool is enabled.

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

expression

Supported Platforms SRX Series, vSRX

Syntax expression *boolean-expression*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain]

Release Information Statement introduced in Junos OS Release 9.3.

Description Configure the Boolean expression. The Boolean expression defines the condition for the individual members of a chain attack that will decide if the chain attack is hit.

For standalone IDP devices, expression overrides order function.

For SRX Series devices, expression and order cannot be configured together. Only one of them can be specified.

Options *boolean-expression*—Boolean operators:

- **or**—If either of the member name patterns match, the expression matches.
- **and**—If both of the member name patterns match, the expression matches. It does not matter which order the members appear in.
- **oand**—If both of the member name patterns match, and if they appear in the same order as in the Boolean Expression, the expression matches.

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

extension-header

Supported Platforms SRX Series

Syntax

```
extension-header {
  destination-option {
    home-address {
      match (equal | greater-than | less-than | not-equal);
      value header-value;
    }
      option-type {
        match (equal | greater-than | less-than | not-equal);
        value header-value;
      }
  }
  routing-header {
    header-type {
      match (equal | greater-than | less-than | not-equal);
      value header-value;
      }
    }
}
```

Hierarchy Level [edit set security idp custom-attack *attack-name* attack-type signature protocol *ipv6*]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv6 extension header.

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

false-positives

Supported Platforms SRX Series, vSRX

Syntax

```
false-positives {
  values [frequently occasionally rarely unknown];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a false positives filter to track attack objects based on the frequency that the attack produces a false positive on your network.

Options **values**—Name of the false positives filter. You can select from the following false positive frequency:

- **frequently**—Frequently track false positive occurrences.
- **occasionally**—Occasionally track false positive occurrences.
- **rarely**—Rarely track false positive occurrences.
- **unknown**—By default, all compound attack objects are set to Unknown. As you fine-tune IDP to your network traffic, you can change this setting to help you track false positives.

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

fifo-max-size (IPS)

Supported Platforms SRX Series, vSRX

Syntax fifo-max-size *value*;

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Sets the maximum IPS FIFO size (range: 1 through 65535).

Options The remaining statements are explained separately. See CLI Explorer.

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

fifo-max-size (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax fifo-max-size *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Sets the maximum FIFO size (range: 1 through 65535).

Options The remaining statements are explained separately. See CLI Explorer.

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

filters

Supported Platforms SRX Series, vSRX

Syntax

```
filters {
  category {
    values [category-value];
  }
  direction {
    expression (and | or);
    values [any client-to-server exclude-any exclude-client-to-server exclude-server-to-client
      server-to-client];
  }
  false-positives {
    values [frequently occasionally rarely unknown];
  }
  performance {
    values [fast normal slow unknown];
  }
  products {
    values [product-value];
  }
  recommended;
  service {
    values [service-value];
  }
  severity {
    values [critical info major minor warning];
  }
  type {
    values [anomaly signature];
  }
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name*]

Release Information Statement introduced in Junos OS Release 9.3. The **expression** option added in Junos OS Release 11.4.

Description To create a dynamic attack group, set the criteria using different types of filters.

Options The remaining statements are explained separately. See CLI Explorer.

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

flow (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
flow {
  (allow-icmp-without-flow | no-allow-icmp-without-flow);
  drop-if-no-policy-loaded;
  drop-on-failover;
  drop-on-limit;
  fifo-max-size value;
  hash-table-size value;
  (log-errors | no-log-errors);
  max-sessions-offset value;
  max-timers-poll-ticks value;
  min-objcache-limit-lt lower-threshold-value;
  min-objcache-limit-ut upper-threshold-value;
  reject-timeout value;
  (reset-on-policy | no-reset-on-policy);
  udp-anticipated-timeout value;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure the IDP engine to manage the packet flow.

Options The remaining statements are explained separately. See CLI Explorer.

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

force-discover (dhcp-client)

Supported Platforms SRX Series

Syntax force-discover ;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* dhcp-client force-discover]

Release Information Statement introduced in Junos OS Release 15.1X49-D80.

Description Forces the DHCP client to send a DHCP discover packet after one to three failed **dhcp-request** attempts. The **force-discover** option ensures that the DHCP server will assign the same or a new IP address to the client.

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

Related Documentation

- *Configuring Optional DHCP Client Attributes*
- *Minimum DHCP Client Configuration*

forwarding-classes (CoS)

Supported Platforms SRX Series, vSRX

Syntax

```
forwarding-classes {
  class class-name {
    priority (high | low);
    queue-num number;
    spu-priority (high | low | medium-high | medium-low);
  }
  queue queue-number {
    class-name {
      priority (high | low);
    }
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 8.5. Statement updated in Junos OS Release 11.4. The **spu-priority** option introduced in Junos OS Release 11.4R2.

Description Configure forwarding classes and assign queue numbers.

Options

- **class class-name**—Display the forwarding class name assigned to the internal queue number.

  NOTE: This option is supported only on SRX1500, SRX5400, SRX5600, and SRX5800.

  NOTE: AppQoS forwarding classes must be different from those defined for interface-based rewriters.

- **priority**—Fabric priority value:
  - **high**—Forwarding class' fabric queuing has high priority.
  - **low**—Forwarding class' fabric queuing has low priority.

  The default **priority** is **low**.

- **queue queue-number**—Specify the internal queue number to which a forwarding class is assigned.
- **spu-priority**—Services Processing Unit (SPU) priority queue, **high**, **medium-high**, **medium-low**, or **low**. The default **spu-priority** is **low**.

NOTE: The **spu-priority** option is only supported on SRX1500 devices and SRX5000 line devices.

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

Related Documentation

- *Example: Configuring AppQoS*

forwarding-process

Supported Platforms SRX Series, vSRX

Syntax

```
forwarding-process {
  application-services {
    enable-gtpu-distribution;
    maximize-alg-sessions;
    maximize-idp-sessions {
      weight (firewall | idp);
    }
    packet-ordering-mode {
      (hardware |software);
    }
  }
}
```

Hierarchy Level [edit security]

Release Information Statement introduced in Junos OS Release 9.6. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX instances.

Description You can configure SRX5400, SRX5600, and SRX5800 devices to switch from an integrated firewall mode to maximize Intrusion Detection and Prevention (IDP) mode to run IDP processing in tap mode and increase the capacity of processing with the **maximize-idp-sessions** option. Inline tap mode can only be configured if the forwarding process mode is set to **maximize-idp-sessions**, which ensures stability and resiliency for firewall services. You also do not need a separate tap or span port to use inline tap mode. When you maximize IDP, you are decoupling IDP processes from firewall processes, allowing the device to support the same number of firewall and IDP sessions, also run the IDP processing in tap mode.

You can configure maximum Application Layer Gateway (ALG) sessions by using the **maximize-alg-sessions** option. By default, the session capacity number for Real-Time Streaming Protocol (RTSP), FTP, and Trivial File Transfer Protocol (TFTP) ALG sessions is 10,000 per flow Services Processing Unit (SPU). You must reboot the device (and its peer in chassis cluster mode) for the configuration to take effect. The **maximize-alg-sessions** option now enables you to increase defaults as follows:

- RTSP, FTP, and TFTP ALG session capacity: 25,000 per flow SPU
- TCP proxy connection capacity: 40,000 per flow SPU

NOTE: Flow session capacity is reduced to half per flow SPU; therefore the aforementioned capacity numbers will not change on central point flow.

Enable GPRS tunneling protocol, user plane(GTP-U) session distribution to distribute GTP-U traffic handled by a Gateway GPRS Support Node (GGSN) and a Serving GPRS Support Node (SGSN) pair on all Services Processing Units (SPUs). You can configure tunnel-base distribution to distribute GTP-U traffic to multiple SPUs by the **enable-gtpu-distribution** option on SRX5400, SRX5600, and SRX5800 devices , which helps to resolve the GTP-U fat session issue. Also, **enable-gtpu-distribution** command is must for enabling stateful GTP-U inspection.

Options The remaining statements are explained separately. See the CLI Explorer.

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

Related Documentation

- application-services (Security Forwarding Process) on page 327
- *Juniper Networks Devices Processing Overview*

from-zone (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax from-zone (*zone-name* | any);

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a source zone to be associated with the security policy. The default value is any.

Options ***zone-name***—Name of the source zone object.

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

global (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
global {
   (enable-all-qmodules | no-enable-all-qmodules);
   (enable-packet-pool | no-enable-packet-pool);
   memory-limit-percent value;
   (policy-lookup-cache | no-policy-lookup-cache);
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure the global rulebase IDP security policy.

Options The remaining statements are explained separately. See CLI Explorer.

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

group-members

Supported Platforms SRX Series, vSRX

Syntax group-members [*attack-or-attack-group-name*];

Hierarchy Level [edit security idp custom-attack-group *custom-attack-group-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the group members in a custom group. The members can be predefined attacks, predefined attack groups, custom attacks, or custom dynamic groups.

Use custom groups for the following tasks:

- To define a specific set of attacks to which you know your network is vulnerable.
- To group your custom attack objects.
- To define a specific set of informational attack objects that you use to keep you aware of what is happening on your network.

Options ***attack-or-attack-group-name***—Name of the attack object or group attack object.

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

hash-table-size (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax hash-table-size *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Sets the packet flow hash table size (range: 1024 through 1,000,000).

Options The remaining statements are explained separately. See CLI Explorer.

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

header-length

Supported Platforms SRX Series, vSRX

Syntax

```
header-length {
  match (equal | greater-than | less-than | not-equal);
  value header-length;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the number of bytes in the TCP header.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***header-length***—Match the number of bytes in the TCP header.

Range: 0 through 15 bytes

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

header-type

Supported Platforms SRX Series

Syntax

```
header-type {
  match (equal | greater-than | less-than | not-equal);
  value header-value;
}
```

Hierarchy Level [edit set security idp custom-attack *attack-name* attack-type signature protocol *ipv6* extension-header routing-header]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv6 routing header type.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value—Match a decimal value.

Range: 0 through 255

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

high-availability (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
high-availability {
  no-policy-cold-synchronization;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configures high availability (chassis cluster) for IDP.

Options The remaining statements are explained separately. See CLI Explorer.

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

home-address

Supported Platforms SRX Series

Syntax

```
home-address {
  match (equal | greater-than | less-than | not-equal);
  value value;
}
```

Hierarchy Level [edit set security idp custom-attack *attack-name* attack-type signature protocol *ipv6* extension-header destination-option]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv6 home address of the mobile node.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value—Match a decimal value.

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

host (Security IDP Sensor Configuration)

Supported Platforms SRX Series, vSRX

Syntax host *ip-address* <port *number*>;

Hierarchy Level [edit security idp sensor-configuration packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the IP address and port number of the server where the packet capture object will be sent.

Options

- **host *ip-address***—The IP address of the server where the packet capture object will be sent.
- **port *number***—The port number of the server where the packet capture object will be sent.

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

icmp (Security IDP Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax icmp;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the attack for the specified ICMP.

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

icmp (Security IDP Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
icmp {
  code {
    match (equal | greater-than | less-than | not-equal);
    value code-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value data-length;
  }
  identification {
    match (equal | greater-than | less-than | not-equal);
    value identification-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  type {
    match (equal | greater-than | less-than | not-equal);
    value type-value;
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the ICMP header information for the signature attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

icmpv6 (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
icmpv6;
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify that the attack is for ICMPv6 packets only.

Required Privilege Level

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

icmpv6 (Security IDP Custom Attack)

Supported Platforms SRX Series

Syntax

```
icmpv6 {
   checksum-validate {
      match (equal | greater-than | less-than | not-equal);
      value checksum-value;
   }
   code {
      match (equal | greater-than | less-than | not-equal);
      value code-value;
   }
   data-length {
      match (equal | greater-than | less-than | not-equal);
      value data-length;
   }
   identification {
      match (equal | greater-than | less-than | not-equal);
      value identification-value;
   }
   sequence-number {
      match (equal | greater-than | less-than | not-equal);
      value sequence-number;
   }
   type {
      match (equal | greater-than | less-than | not-equal);
      value type-value;
   }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Allow IDP to match the attack for the specified ICMPv6.

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

identification (Security ICMP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
identification {
   match (equal | greater-than | less-than | not-equal);
   value identification-value;
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type signature protocol icmp]
[edit security idp custom-attack *attack-name* attack-type signature protocol icmpv6]

Release Information Statement introduced in Junos OS Release 9.3. Statement modified in Junos OS Release 12.3X48-D25 to add ICMPv6 protocol support.

Description Specify a unique value used by the destination system to associate requests and replies.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***identification-value***—Match a decimal value.

Range: 0 through 65,535

Required Privilege Level

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

identification (Security IP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
identification {
  match (equal | greater-than | less-than | not-equal);
  value identification-value;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a unique value used by the destination system to reassemble a fragmented packet.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***identification-value***—Match a decimal value.

Range: 0 through 65,535

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

idp (Application Services)

Supported Platforms SRX Series, vSRX

Syntax idp;

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit application-services]

Release Information Statement introduced in Junos OS Release 11.1.

Description Configure Intrusion Detection and Prevention (IDP) for application services.

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

idp (Security Alarms)

Supported Platforms SRX Series, vSRX

Syntax idp;

Hierarchy Level [edit security alarms potential-violation]

Release Information Statement introduced in Junos OS Release 11.2.

Description Configure alarms for IDP attack.

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

idp-policy (Security)

Supported Platforms SRX Series, vSRX

Syntax

```
idp-policy policy-name {
  rulebase-exempt {
    rule rule-name {
      description text;
      match {
        attacks {
          custom-attack-groups [attack-group-name];
          custom-attacks [attack-name];
          dynamic-attack-groups [attack-group-name];
          predefined-attack-groups [attack-group-name];
          predefined-attacks [attack-name];
        }
        destination-address ([address-name] | any | any-ipv4 | any-ipv6);
        destination-except [address-name];
        from-zone (zone-name | any );
        source-address ([address-name] | any | any-ipv4 | any-ipv6);
        source-except [address-name];
        to-zone (zone-name | any);
      }
    }
  }
  rulebase-ips {
    rule rule-name {
      description text;
      match {
        application (application-name | any | default);
        attacks {
          custom-attack-groups [attack-group-name];
          custom-attacks [attack-name];
          dynamic-attack-groups [attack-group-name];
          predefined-attack-groups [attack-group-name];
          predefined-attacks [attack-name];
        }
        destination-address ([address-name] | any | any-ipv4 | any-ipv6);
        destination-except [address-name];
        from-zone (zone-name | any );
        source-address ([address-name] | any | any-ipv4 | any-ipv6);
        source-except [address-name];
        to-zone (zone-name | any);
      }
      terminal;
      then {
        action {
          class-of-service {
            dscp-code-point number;
            forwarding-class forwarding-class;
          }
          (close-client | close-client-and-server | close-server |drop-connection | drop-packet
            | ignore-connection | mark-diffserv value | no-action | recommended);
        }
        ip-action {
          (ip-block | ip-close | ip-notify);
          log;
          log-create;
          refresh-timeout;
          target (destination-address | service | source-address | source-zone |
            source-zone-address | zone-service);
          timeout seconds;
        }
        notification {
          log-attacks {
            alert;
          }
          packet-log {
            post-attack number;
            post-attack-timeout seconds;
            pre-attack number;
          }
        }
        severity (critical | info | major | minor | warning);
      }
    }
  }
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure a security IDP policy.

Options ***policy-name***—Name of the IDP policy.

The remaining statements are explained separately. See CLI Explorer.

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

ignore-memory-overflow

Supported Platforms SRX Series, vSRX

Syntax (ignore-memory-overflow | no-ignore-memory-overflow);

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable the TCP reassembler to ignore the memory overflow to prevent the dropping of IDP custom applications. By default this feature is enabled.

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

ignore-reassembly-memory-overflow | no-ignore-reassembly-memory-overflow

Supported Platforms SRX Series, vSRX

Syntax (ignore-reassembly-memory-overflow | no-ignore-reassembly-memory-overflow);

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 9.2.

Description Reassembly memory overflow occurs when the memory allocated for the reassembly of TCP fragments is exceeded. When the reassembly of TCP fragments exceeds the memory limit, defined with **max-packet-mem-ratio**, you can define the system behavior to ignore or drop the offending packets. If the **ignore-reassembly-memory-overflow** command is enabled on the SRX device, IDP will ignore and permit packets from sessions which trigger a reassembly memory overflow. If you enable the **no-ignore-reassembly-memory-overflow** command when reassembly memory overflow occurs, packets of that session are dropped by the device. By default, the **ignore-reassembly-memory-overflow** command is enabled.

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

Related Documentation

- max-packet-mem-ratio on page 418

ignore-reassembly-overflow

Supported Platforms SRX Series, vSRX

Syntax ignore-reassembly-overflow

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 9.3.

Description Enable the TCP reassembler to ignore the global reassembly overflow to prevent the dropping of application traffic. This feature is enabled by default.

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

ignore-regular-expression

Supported Platforms SRX Series, vSRX

Syntax (ignore-regular-expression | no-ignore-regular-expression);

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description To detect intrusion attempts, you can enable regular expression by issuing the **no-ignore-regular-expression** command. By default, the **no-ignore-regular-expression** command is enabled. If you specify the **ignore-regular-expression** command, regular expression pattern matching will be disabled when detecting intrusion attempts.

Default Regular expression is enabled by default.

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

ihl (Security IDP Custom Attack)

Supported Platforms SRX Series

Syntax

```
ihl {
  match (equal | greater-than | less-than | not-equal);
  value ihl-value;
}
```

Hierarchy Level [edit set security idp custom-attack *ipv4_custom* attack-type signature protocol *ipv4*]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv4 header length in words.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value—Match a decimal value.

Range: 0 through 15

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

include-destination-address

Supported Platforms SRX Series, vSRX

Syntax

```
(include-destination-address | no-include-destination-address);
```

Hierarchy Level [edit security idp sensor-configuration log suppression]

Release Information Statement introduced in Junos OS Release 9.2.

Description When log suppression is enabled, multiple occurrences of events with the same source, service, and matching attack object generate a single log record with a count of occurrences. If you enable this option, log suppression will only combine log records for events with a matching source as well. The IDP Sensor does not consider destination when determining matching events for log suppression. By default this setting is disabled.

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

install

Supported Platforms SRX Series, vSRX

Syntax

```
install {
   ignore-version-check;
}
```

Hierarchy Level [edit security idp security-package]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configures the **install** command.

Options The remaining statements are explained separately. See CLI Explorer.

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

interfaces (CoS)

Syntax

```
interfaces
  interface-name {
    input-scheduler-map map-name ;
    input-shaping-rate rate ;
    scheduler-map  map-name ;
    scheduler-map-chassis  map-name  ;
    shaping-rate rate ;
    unit  logical-unit-number  {
      adaptive-shaper  adaptive-shaper-name  ;
      classifiers {
        (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence)
        ( classifier-name  | default);
      }
      forwarding-class  class-name  ;
      fragmentation-map map-name ;
      input-scheduler-map map-name  ;
      input-shaping-rate (percent  percentage  |  rate  );
      input-traffic-control-profile  profiler-name  shared-instance  instance-name  ;
      loss-priority-maps {
        default;
        map-name ;
      }
      output-traffic-control-profile  profile-name  shared-instance  instance-name  ;
      rewrite-rules {
        dscp (  rewrite-name  | default);
        dscp-ipv6 (  rewrite-name  | default);
        exp (  rewrite-name  | default) protocol  protocol-types  ;
        frame-relay-de (  rewrite-name  | default);
        inet-precedence (  rewrite-name  | default);
      }
      scheduler-map map-name ;
      shaping-rate rate  ;
      virtual-channel-group group-name  ;
    }
  }
}
```

Hierarchy Level

[edit class-of-service interface *interface-name* unit *number*]

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Associate the class-of-service configuration elements with an interface.

Options

interface *interface-name* unit *number*—The user-specified interface name and unit number.

Required Privilege Level

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

Related Documentation

- *Class of Service Feature Guide for Security Devices*

interval (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
interval hours;
```

Hierarchy Level [edit security idp security-package automatic]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the amount of time that the device waits before updating the signature database. User should insert a default value.

Options *hours*—Number of hours that the device waits.

Range: 24 through 336 hours

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

ip (Security IDP Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
ip {
   protocol-number transport-layer-protocol-number;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the attack for a specified IP protocol type.

Options **protocol-number** ***transport-layer-protocol-number***—Transport Layer protocol number.

Range: 0 through 139

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

ip-action (Security IDP Rulebase IPS)

Supported Platforms SRX Series, vSRX

Syntax

```
ip-action {
  (ip-block | ip-close | ip-notify);
  log;
  log-create;
  refresh-timeout;
  target (destination-address | service | source-address | source-zone | source-zone-address
    | zone-service);
  timeout seconds;
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the actions you want IDP to take against future connections that use the same IP address.

Options The remaining statements are explained separately. See CLI Explorer.

NOTE: For ICMP flows, the destination port is 0; therefore, any ICMP flow matching source port, source address, and destination address is blocked.

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

ip-block

Supported Platforms SRX Series, vSRX

Syntax ip-block;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Block future connections of any session that matches the IP action. If there is an IP action match with multiple rules, then the most severe IP action of all the matched rules is applied. The highest IP action priority (that is, the most severe action) is Drop/Block, then Close, then Notify.

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

ip-close

Supported Platforms SRX Series, vSRX

Syntax ip-close;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Close future connections of any new sessions that match the IP action by sending RST packets to the client and server.

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

ip-connection-rate-limit

Supported Platforms SRX Series, vSRX

Syntax ip-connection-rate-limit *connections-per-second*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ddos rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 10.2.

Description When a match is made in a rulebase-ddos rule you can set the **then** action to ip-connection-rate-limit, which will limit the rate of future connections based on a connections per second limit that you set. This can be used to reduce the number of attacks from a client.

Options *value*—Defines the connection rate limit per second on the matched host.

Range: 1 to the maximum connections per second capability of the device.

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

ip-flags

Supported Platforms SRX Series, vSRX

Syntax

```
ip-flags {
  (df | no-df);
  (mf | no-mf);
  (rb | no-rb);
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify that IDP looks for a pattern match whether or not the IP flag is set.

Options

- **df | no-df**—When set, the df (Don't Fragment) indicates that the packet cannot be fragmented for transmission. When unset, it indicates that the packet can be fragmented.
- **mf | no-mf**—When set, the mf (More Fragments) indicates that the packet contains more fragments. When unset, it indicates that no more fragments remain.
- **rb | no-rb**—When set, the rb (Reserved Bit) indicates that the bit is reserved.

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

ip-notify

Supported Platforms SRX Series, vSRX

Syntax

```
ip-notify;
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Do not take any action against future traffic, but do log the event.

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

ips

Supported Platforms SRX Series, vSRX

Syntax

```
ips {
  content-decompression-max-memory-kb value;
  content-decompression-max-ratio value;
  (detect-shellcode | no-detect-shellcode);
  fifo-max-size value;
  (ignore-regular-expression | no-ignore-regular-expression);
  log-supercede-min minimum-value;
  pre-filter-shellcode;
  (process-ignore-s2c | no-process-ignore-s2c);
  (process-override | no-process-override);
  process-port port-number;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure IPS security policy sensor settings.

Options The remaining statements are explained separately. See CLI Explorer.

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

ipv4 (Security IDP Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
ipv4 {
  destination {
    match (equal | greater-than | less-than | not-equal);
    value ip-address-or-hostname;
  }
  identification {
    match (equal | greater-than | less-than | not-equal);
    value identification-value;
  }
  ip-flags {
    (df | no-df);
    (mf | no-mf);
    (rb | no-rb);
  }
  protocol {
    match (equal | greater-than | less-than | not-equal);
    value transport-layer-protocol-id;
  }
  source {
    match (equal | greater-than | less-than | not-equal);
    value ip-address-or-hostname;
  }
  tos {
    match (equal | greater-than | less-than | not-equal);
    value type-of-service-in-decimal;
  }
  total-length {
    match (equal | greater-than | less-than | not-equal);
    value total-length-of-ip-datagram;
  }
  ttl {
    match (equal | greater-than | less-than | not-equal);
    value time-to-live;
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the IP header information for the signature attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

key-exchange

Supported Platforms MX Series, PTX Series, SRX Series, vSRX

Syntax key-exchange [*algorithm1 algorithm2*...];

Hierarchy Level [edit system services ssh]

Release Information Statement introduced in Junos OS Release 11.2. Support for curve25519-sha256 added in Junos OS Release 12.1X47-D10.

Description Specify the set of Diffie-Hellman key exchange methods that the SSH server can use.

Options One or more of the following Diffie-Hellman key exchange methods:

- **curve25519-sha256**—The EC Diffie-Hellman key exchange method on Curve25519 with SHA2-256.
- **dh-group1-sha1**—The Diffie-Hellman group1 algorithm using SHA-1.
- **dh-group14-sha1**—The Diffie-Hellman group14 algorithm using SHA-1.
- **ecdh-sha2-nistp256**—The ECDH key exchange method with ephemeral keys generated on the nistp256 curve.
- **ecdh-sha2-nistp384**—The ECDH key exchange method with ephemeral keys generated on the nistp384 curve.
- **ecdh-sha2-nistp521**—The ECDH key exchange method with ephemeral keys generated on the nistp521 curve.
- **group-exchange-sha1**—The group exchange algorithm using SHA-1.
- **group-exchange-sha2**—The group exchange algorithm using SHA-2.

NOTE: The key-exchange represents a set. To configure key-exchange:

```
user@host#set system services ssh key-exchange [ecdh-sha2-nistp256
   group-exchange-sha1]
```

NOTE: Table 95 on page 410 shows the supportability of Diffie-Hellman key exchange methods on FIPS mode.

Table 95: Supportability of Diffie-Hellman key exchange methods on FIPS mode

Diffie-Hellman key exchange methods	Supported on FIPS mode
curve25519-sha256	No
dh-group1-sha1	No
dh-group14-sha1	Yes
ecdh-sha2-nistp256	Yes
ecdh-sha2-nistp384	Yes
ecdh-sha2-nistp521	Yes
group-exchange-sha1	No
group-exchange-sha2	No

Required Privilege Level

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

Related Documentation

- *Configuring SSH Service for Remote Access to the Router or Switch*
- *ciphers*
- *macs*

key-protection (Security IDP)

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax key-protection;

Hierarchy Level [edit security idp sensor-configuration ssl-inspection]

Release Information Statement introduced in Junos OS Release 11.2.

Description Enabling key protection provides improved security. When key protection is enabled, persistent keys are encrypted when not in use.

Enabling or disabling of this option requires rebooting the device.

Required Privilege Level

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

key-protection (Security IDP Sensor Configuration)

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
key-protection;
```

Hierarchy Level [edit security idp sensor-configuration ssl-inspection]

Release Information Statement introduced in Junos OS Release 11.2.

Description Enable secure key handling. This option is off by default.

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

log (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
log {
  cache-size size;
  suppression {
    disable;
    (include-destination-address | no-include-destination-address);
    max-logs-operate value;
    max-time-report value;
    start-log value;
  }
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure IDP security policy logs.

Options The remaining statements are explained separately. See CLI Explorer.

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

log (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax

```
log;
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Log the information about the IP action against the traffic that matches a rule.

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

log-attacks

Supported Platforms SRX Series, vSRX

Syntax

```
log-attacks {
   alert;
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable the log attacks to create a log record that appears in the log viewer.

Options The remaining statements are explained separately. See CLI Explorer.

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

log-create

Supported Platforms SRX Series, vSRX

Syntax log-create;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 10.2.

Description Generate a log event on installing the ip-action filter.

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

log-errors

Supported Platforms SRX Series, vSRX

Syntax (log-errors | no-log-errors);

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable the error log to generate the result of success or failure about the flow. A flow-related error is when IDP receives a packet that does not fit into the expected flow. By default an error log is enabled.

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

log-supercede-min

Supported Platforms SRX Series, vSRX

Syntax

```
log-supercede-min minimum-value;
```

Hierarchy Level

```
[edit security idp sensor-configuration ips]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the amount of time to supersede the IPS sensor logs.

Options *minimum-value*—Minimum time to supersede the log.

Range: 0 through 65,535 seconds

Default: 1 second

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

loss-priority (CoS Rewrite Rules)

Supported Platforms SRX Series, vSRX

Syntax loss-priority *level*;

Hierarchy Level [edit class-of-service rewrite-rules *type rewrite-name* forwarding-class *class-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a loss priority to which to apply a rewrite rule. The rewrite rule sets the code-point aliases and bit patterns for a specific forwarding class and packet loss priority (PLP). The inputs for the map are the forwarding class and the PLP. The output of the map is the code-point alias or bit pattern.

Options *level* can be one of the following:

- **high**—The rewrite rule applies to packets with high loss priority.
- **low**—The rewrite rule applies to packets with low loss priority.
- **medium-high**—The rewrite rule applies to packets with medium-high loss priority.
- **medium-low**—The rewrite rule applies to packets with medium-low loss priority.

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

Related Documentation

- *Class of Service Feature Guide for Security Devices*

match (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax

```
match {
  attacks {
    custom-attack-groups [attack-group-name];
    custom-attacks [attack-name];
    dynamic-attack-groups [attack-group-name];
    predefined-attack-groups [attack-group-name];
    predefined-attacks [attack-name];
  }
  destination-address ([address-name] | any | any-ipv4 | any-ipv6);
  destination-except [address-name];
  from-zone (zone-name | any );
  source-address ([address-name] | any | any-ipv4 | any-ipv6);
  source-except [address-name];
  to-zone (zone-name | any);
}
```

Hierarchy Level

[edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name*]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the rules to be used as match criteria.

Options The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

max-flow-mem

Supported Platforms SRX Series, vSRX

Syntax max-flow-mem *value*;

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 9.2.

Description Define the maximum TCP flow memory that the IDP sensor can handle.

Options *value*—Maximum TCP flow memory in kilobytes.

Range: 64 through 4,294,967,295 kilobytes

Default: 1024 kilobytes

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

max-logs-operate

Supported Platforms SRX Series, vSRX

Syntax max-logs-operate *value*;

Hierarchy Level [edit security idp sensor-configuration log suppression]

Release Information Statement introduced in Junos OS Release 9.2.

Description When log suppression is enabled, IDP must cache log records so that it can identify when multiple occurrences of the same event occur. This setting specifies how many log records are tracked simultaneously by IDP.

Options *value*—Maximum number of log records are tracked by IDP.

Range: 256 through 65,536 records

Default: 16,384 records

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

max-packet-mem-ratio

Supported Platforms SRX Series, vSRX

Syntax max-packet-mem-ratio *percentage-value*;

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description By default, values for IDP reassembler packet memory are established as percentages of all memory. In most cases, these default values are adequate.

If a deployment exhibits an excessive number of dropped TCP packets or retransmissions resulting in high IDP reassembly memory usage, use the **max-packet-mem-ratio** option to reset the percentage of available IDP memory for IDP reassembly packet memory. Acceptable values are between 5 percent and 40 percent.

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

max-packet-memory-ratio

Supported Platforms SRX Series, vSRX

Syntax max-packet-memory-ratio *percentage-value*;

Hierarchy Level [edit security idp sensor-configuration application-identification]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description By default, the amount of IDP memory used for application identification packet memory is established as a percentage of all IDP memory. In most cases, the default value is adequate.

If a deployment exhibits an excessive number of ignored IDP sessions due to application identification memory allocation failures, use the **max-packet-memory-ratio** option to set application identification packet memory limit at a higher percentage of available IDP memory. This memory is only used by IDP in cases where application identification delays identifying an application. Acceptable values are between 5 percent and 40 percent.

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

max-reass-packet-memory-ratio

Supported Platforms SRX Series, vSRX

Syntax max-reass-packet-memory-ratio *percentage-value*;

Hierarchy Level [edit security idp sensor-configuration application-identification]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description By default, the amount of IDP memory used for packet memory by the application identification reassembler is established as a percentage of all IDP memory. In most cases, the default value is adequate.

If a deployment exhibits an excessive number of ignored IDP sessions due to packet memory limitations of the application identification reassembler, use the **max-reass-packet-memory-ratio** option to set the reassembler packet memory limit to a higher percentage of available IDP memory. Acceptable values are between 5% and 40%.

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

max-sessions (Security Packet Log)

Supported Platforms SRX Series, vSRX

Syntax max-sessions *percentage*;

Hierarchy Level [edit security idp sensor-configuration packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the maximum number of sessions actively conducting pre-attack packet captures on a device at one time. This value is expressed as a percentage of the maximum number of IDP sessions for the device.

Options *percentage*—Maximum number of packet capture sessions expressed as a percentage of the IDP session capacity for the device.

Range: 1 through 100 percent

Default: 10

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

max-sessions-offset (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax max-sessions-offset *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set an offset (percentage) for the maximum IDP session limit. The **max-sessions-offset** option sets an offset for the maximum IDP session limit. When the number of IDP sessions exceeds the maximum session limit, a warning is logged that conditions exist where IDP sessions could be dropped. When the number of IDP sessions drops below the maximum IDP session limit minus the offset value, a message is logged that conditions have returned to normal.

Options **value**—Maximum session offset limit percentage is 0 through 99.

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

max-synacks-queued

Supported Platforms SRX Series, vSRX

Syntax max-synacks-queued *value*;

Hierarchy Level [edit security idp sensor-configuration re-assembler]

Release Information Statement introduced in Junos OS Release 12.1X46-D25.

Description Define the maximum limit for queuing Syn/Ack packets with different SEQ numbers.

Options ***value***—Maximum synchronization acknowledgements queued with different SEQ numbers.

Range: 0 through 5

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

max-tcp-session-packet-memory

Supported Platforms SRX Series, vSRX

Syntax max-tcp-session-packet-memory *value*;

Hierarchy Level [edit security idp sensor-configuration application-identification]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the maximum number of TCP sessions that IDP maintains. If the sensor reaches the maximum, it drops all new TCP sessions.

Options *value*—Maximum number of TCP sessions.

Range: 0 through 60,000

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

max-time-report

Supported Platforms SRX Series, vSRX

Syntax max-time-report *value*;

Hierarchy Level [edit security idp sensor-configuration log suppression]

Release Information Statement introduced in Junos OS Release 9.2.

Description When log suppression is enabled, IDP maintains a count of multiple occurrences of the same event. After the specified number of seconds has passed, IDP writes a single log entry containing the count of occurrences.

Options *value*—Time after which IDP writes a single log entry containing the count of occurrences.

Range: 1 through 60 seconds

Default: 5 seconds

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

max-timers-poll-ticks

Supported Platforms SRX Series, vSRX

Syntax max-timers-poll-ticks *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the time at which timer ticks at regular interval.

Options *value*—Maximum amount of time at which the timer ticks.

Range: 0 through 1000 ticks

Default: 1000 ticks

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

max-udp-session-packet-memory

Supported Platforms SRX Series, vSRX

Syntax max-udp-session-packet-memory *value*;

Hierarchy Level [edit security idp sensor-configuration application-identification]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the maximum number of UDP sessions that IDP maintains. If the sensor reaches the maximum, it drops all new UDP sessions.

Options *value*—Maximum number of UDP sessions.

Range: 0 through 20,000

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

maximize-idp-sessions

Supported Platforms SRX Series

Syntax

```
maximize-idp-sessions {
  weight (equal | firewall | idp);
}
```

Hierarchy Level [edit security forwarding-process application-services]

Release Information Statement introduced in Junos OS Release 9.6.

Description If you are deploying IDP policies, you can tune the device to increase IDP session capacity. By using the provided commands to change the way the system allocates resources, you can achieve a higher IDP session capacity. See weight for information about the options provided.

This statement is supported on SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

NOTE: The IDP session capacity is restricted to 100,000 sessions per SPU.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Juniper Networks Devices Processing Overview*

maximum-cache-size

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
maximum-cache-size number;
```

Hierarchy Level [edit security idp sensor-configuration ssl-inspection]

Release Information Statement introduced in Junos OS Release 10.2.

Description Maximum SSL session ID cache size.

Options *maximum-cache-size*—Maximum number of SSL session ID cache size.

Range: 1 through 5,000,000 sessions

Default: 5,000,000

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

member (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
member member-name {
  attack-type {
    (anomaly ...same statements as in [edit security idp custom-attack attack-name
      attack-type anomaly] hierarchy level | signature ...same statements as in [edit security
      idp custom-attack attack-name attack-type signature] hierarchy level);
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain]

Release Information Statement introduced in Junos OS Release 9.3.

Description Create the list of member attacks.

Options *member-name*—Name of the member list.

The remaining statements are explained separately. See CLI Explorer.

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

min-objcache-limit-lt

Supported Platforms SRX Series, vSRX

Syntax min-objcache-limit-lt *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description Memory lower threshold limit percentage.

Options **value**— Memory lower threshold limit percentage.

percentage range—1 through 100

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

min-objcache-limit-ut

Supported Platforms SRX Series, vSRX

Syntax min-objcache-limit-ut *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 12.1X44-D20.

Description Memory upper threshold limit percentage.

Options **value**—Memory upper threshold limit percentage.

percentage range— 1 through 100

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

mss (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
mss {
   match (equal | greater-than | less-than | not-equal);
   value maximum-segment-size;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the maximum segment size (MSS) in the TCP header.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***maximum-segment-size***—Match the maximum segment size value.

Range: 0 through 65,535

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

negate

Supported Platforms SRX Series, vSRX

Syntax negate;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Select negate to exclude the specified pattern from being matched.

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

nested-application (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax nested-application *nested-application-name*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the nested application name.

Options *nested-application-name*—Name of the nested application.

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

no-recommended

Supported Platforms SRX Series, vSRX

Syntax no-recommended;

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 11.4R6.

Description Specify non recommended attack objects in the dynamic attack group.

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

Related Documentation

- Understanding IDP Policy Rules on page 55

notification

Supported Platforms SRX Series, vSRX

Syntax

```
notification {
   log-attacks {
      alert;
   }
   packet-log {
      post-attack number;
      post-attack-timeout seconds;
      pre-attack number;
   }
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then]

Release Information Statement introduced in Junos OS Release 9.2. Added packet capture support in Junos OS Release 10.2.

Description Configure the logging options against the action. When attacks are detected, you can choose to log an attack and create log records with attack information and send that information to the log server.

Options The remaining statements are explained separately. See CLI Explorer.

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

option (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
option {
   match (equal | greater-than | less-than | not-equal);
   value tcp-option;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the TCP option type (kind field in the TCP header).

Options **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.

value ***tcp-option***—Match the option value.

Range: 0 through 255

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

option-type

Supported Platforms SRX Series

Syntax

```
option-type {
   match (equal | greater-than | less-than | not-equal);
   value header-value;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol *ipv6* extension-header destination-option]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the type of option for destination header type.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value—Match a decimal value.

Range: 0 through 255

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

order (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
order;
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain]

Release Information Statement introduced in Junos OS Release 9.3.

Description Create a compound attack object that must match each member signature or protocol anomaly in the order you specify. If you do not specify an ordered match, the compound attack object still must match all members, but the attacks or protocol anomalies can appear in random order.

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

packet-log (Security IDP Policy)

Supported Platforms SRX1500, SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
packet-log {
   post-attack number;
   post-attack-timeout seconds;
   pre-attack number;
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification]

Release Information Statement introduced in Junos OS Release 10.2.

Description In response to a rule match, capture the packets received before and after the attack for further offline analysis of attacker behavior. You can configure the number of pre-attack and post-attack packets to be captured for this attack, and limit the duration of post-attack packet capture by specifying a timeout value.

Options The remaining statements are explained separately. See CLI Explorer.

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

packet-log (Security IDP Sensor Configuration)

Supported Platforms SRX Series, vSRX

Syntax

```
packet-log {
  host ip-address <port number>;
  max-sessions percentage;
  source-address ip-address;
  total-memory percentage;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the sensor for packet capture. This configuration defines the amount of memory to be allocated for packet capture and the maximum number of sessions that can generate packet capture data for the device at one time. The configuration also identifies the source address and host address for transmission of the completed packet capture object.

Options The remaining statements are explained separately. See CLI Explorer.

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

pattern (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax pattern *signature-pattern*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the pattern IDP should match. You construct the attack pattern just as you would when creating a new signature attack object.

Options *signature-pattern*—Specify the signature pattern.

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

pattern-pcre (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
pattern-pcre signature-pattern-pcre;
```

Hierarchy Level

```
[edit security idp custom-attack attack-name attack-type signature]
```

Release Information Statement introduced in Junos OS Release 15.1x49-D40.

Description Specify the pattern in standard PCRE format. You construct the attack pattern in PCRE format just as you would when creating a new signature attack object. This is an optional field. The pattern field is unused under this configuration.

Options ***signature-pattern-pcre*** —Specify the signature pattern in standard PCRE format.

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

performance

Supported Platforms SRX Series, vSRX

Syntax

```
performance {
  values [fast normal slow unknown];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a performance filter to add attack objects based on the performance level that is vulnerable to the attack.

Options **values**—Name of the performance filter. You can select from the following performance levels:

- **fast**—Fast track performance level.
- **normal**—Normal track performance level.
- **slow**—Slow track performance level.
- **unknown**—By default, all compound attack objects are set to Unknown. As you fine-tune IDP to your network traffic, you can change this setting to help you track performance level.

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

permit (Security Policies)

Supported Platforms SRX Series, vSRX

Syntax

```
permit {
   application-services {
      application-firewall {
         rule-set rule-set-name;
      }
      application-traffic-control {
         rule-set rule-set-name;
      }
      gprs-gtp-profile profile-name;
      gprs-sctp-profile profile-name;
      idp;
      redirect-wx | reverse-redirect-wx;
      ssl-proxy {
         profile-name profile-name;
      }
      uac-policy {
         captive-portal captive-portal;
      }
      utm-policy policy-name;
   }
   destination-address {
      drop-translated;
      drop-untranslated;
   }
   firewall-authentication {
      pass-through {
         access-profile profile-name;
         client-match user-or-group-name;
         ssl-termination-profile profile-name;
         web-redirect;
         web-redirect-to-https;
      }
      user-firewall {
         access-profile profile-name;
         domain domain-name
         ssl-termination-profile profile-name;
      }
      web-authentication {
         client-match user-or-group-name;
      }
   }
   services-offload;
   tcp-options {
      sequence-check-required;
      syn-check-required;
   }
   tunnel {
      ipsec-group-vpn group-vpn;
      ipsec-vpn vpn-name;
      pair-policy pair-policy;
```

```
  }
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **tcp-options** added in Junos OS Release 10.4. Support for the **services-offload** option added in Junos OS Release 11.4. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10.

Description Specify the policy action to perform when packets match the defined criteria.

Options The remaining statements are explained separately. See CLI Explorer.

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

policy-lookup-cache

Supported Platforms SRX Series, vSRX

Syntax (policy-lookup-cache | no-policy-lookup-cache);

Hierarchy Level [edit security idp sensor-configuration global]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable cache to accelerate IDP policy lookup which improves IDP performance.

Default **policy-lookup-cache** is enabled by default.

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

post-attack

Supported Platforms SRX Series, vSRX

Syntax post-attack *number*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify the number of packets received after an attack that should be captured for further analysis of attacker behavior. If post-attack packets are not significant to your analysis or the configured attack response ends packet transfer, you can set the post-attack option to 0.

Options *number*—Number of post-attack packets to be captured.

Range: 0 through 255

Default: 1

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

post-attack-timeout

Supported Platforms SRX Series, vSRX

Syntax post-attack-timeout *seconds*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify a time limit for capturing post-attack packets for a session. No packet capture is conducted after the timeout has expired.

Options *seconds*—Maximum number of seconds for post-attack packet capture.

Range: 0 through 1800 seconds

Default: 5

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

potential-violation

Supported Platforms SRX Series, vSRX

Syntax

```
potential-violation {
   authentication failures;
   cryptographic-self-test;
   decryption-failures {
      threshold value;
   }
   encryption-failures {
      threshold value;
   }
   idp;
   ike-phase1-failures {
      threshold value;
   }
   ike-phase2-failures {
      threshold value;
   }
   key-generation-self-test;
   non-cryptographic-self-test;
   policy {
      application {
         duration interval;
         size count;
         threshold value;
      }
      destination-ip {
         duration interval;
         size count;
         threshold value;
      }
      policy match {
         duration interval;
         size count;
         threshold value;
      }
      source-ip {
         duration interval;
         size count;
         threshold value;
      }
   }
   replay-attacks {
      threshold value;
   }
   security-log-percent-full percentage;
}
```

Hierarchy Level [edit security alarms]

Release Information Statement introduced in Junos OS Release 11.2.

Description Configure alarms for potential violation.

Options The remaining statements are explained separately. See CLI Explorer.

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

pre-attack

Supported Platforms SRX Series, vSRX

Syntax pre-attack *number*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then notification packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify the number of packets received before an attack that should be captured for further analysis of attacker behavior.

Options *number*—Number of pre-attack packets.

Range: 1 through 255

Default: 1

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

pre-filter-shellcode

Supported Platforms SRX Series, vSRX

Syntax pre-filter-shellcode;

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable to pre-filter the shell code and protects it from buffer overflow attacks. By default this setting is enabled.

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

predefined-attack-groups

Supported Platforms SRX Series, vSRX

Syntax predefined-attack-groups [*attack-group-name*];

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match attacks],
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match attacks]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify predefined attack groups that you can use to match the traffic against known attack objects. You can update only the list of attack objects.

Options ***attack-name*** —Name of the predefined attack object group.

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

predefined-attacks

Supported Platforms SRX Series, vSRX

Syntax predefined-attacks [*attack-name*];

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match attacks],
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match attacks]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify predefined attack objects that you can use to match the traffic against known attacks. You can update only the list of attack objects.

Options ***attack-name***—Name of the predefined attack objects.

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

process-ignore-s2c

Supported Platforms SRX Series, vSRX

Syntax (process-ignore-s2c | no-process-ignore-s2c);

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set the command to disable the server-to-client inspection.

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

process-override

Supported Platforms SRX Series, vSRX

Syntax (process-override | no-process-override);

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set the command to forcefully run the IDS inspection module even if there is no policy match.

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

process-port

Supported Platforms SRX Series, vSRX

Syntax

```
process-port port-number;
```

Hierarchy Level [edit security idp sensor-configuration ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set the command to a specific port to forcefully run the IDS inspection module on that TCP/UDP port even if there is no policy match.

Options *port-number*—Port number.

Range: 0 through 65,535

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

products

Supported Platforms SRX Series, vSRX

Syntax

```
products {
  values [product-value];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a products filter to add attack objects based on the application that is vulnerable to the attack.

Options **values**—Name of the products filter. You can configure multiple filters separated by spaces and enclosed in square brackets.

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

protocol (Security IDP IP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
protocol {
   match (equal | greater-than | less-than | not-equal);
   value transport-layer-protocol-id;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the Transport Layer protocol number.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***transport-layer-protocol-id***—Match the Transport Layer protocol ID.

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

protocol (Security IDP Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
protocol {
  icmp {
    checksum-validate {
      match (equal | greater-than | less-than | not-equal);
      value checksum-value;
    }
    code {
      match (equal | greater-than | less-than | not-equal);
      value code-value;
    }
    data-length {
      match (equal | greater-than | less-than | not-equal);
      value data-length;
    }
    identification {
      match (equal | greater-than | less-than | not-equal);
      value identification-value;
    }
    sequence-number {
      match (equal | greater-than | less-than | not-equal);
      value sequence-number;
    }
    type {
      match (equal | greater-than | less-than | not-equal);
      value type-value;
    }
  }
  icmpv6 {
    checksum-validate {
      match (equal | greater-than | less-than | not-equal);
      value checksum-value;
    }
    code {
      match (equal | greater-than | less-than | not-equal);
      value code-value;
    }
    data-length {
      match (equal | greater-than | less-than | not-equal);
      value data-length;
    }
    identification {
      match (equal | greater-than | less-than | not-equal);
      value identification-value;
    }
    sequence-number {
      match (equal | greater-than | less-than | not-equal);
      value sequence-number;
    }
    type {
      match (equal | greater-than | less-than | not-equal);
      value type-value;
    }
  }
  ipv4 {
    checksum-validate {
      match (equal | greater-than | less-than | not-equal);
      value checksum-value;
    }
    destination {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
    }
    identification {
      match (equal | greater-than | less-than | not-equal);
      value identification-value;
    }
    ihl {
      match (equal | greater-than | less-than | not-equal);
      value ihl-value;
    }
    ip-flags {
      (df | no-df);
      (mf | no-mf);
      (rb | no-rb);
    }
    protocol {
      match (equal | greater-than | less-than | not-equal);
      value transport-layer-protocol-id;
    }
    source {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
    }
    tos {
      match (equal | greater-than | less-than | not-equal);
      value type-of-service-in-decimal;
    }
    total-length {
      match (equal | greater-than | less-than | not-equal);
      value total-length-of-ip-datagram;
    }
    ttl {
      match (equal | greater-than | less-than | not-equal);
      value time-to-live;
    }
  }
  ipv6 {
    destination {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
    }
    extension-header {
      destination-option {
        home-address {
          match (equal | greater-than | less-than | not-equal);
          value header-value;
```

```
            }
                option-type {
                    match (equal | greater-than | less-than | not-equal);
                    value header-value;
                }
        }
        routing-header {
            header-type {
                match (equal | greater-than | less-than | not-equal);
                value header-value;
                }
            }
    }
    flow-label {
        match (equal | greater-than | less-than | not-equal);
        value flow-label-value;
    }
    hop-limit {
        match (equal | greater-than | less-than | not-equal);
        value hop-limit-value;
    }
    next-header {
        match (equal | greater-than | less-than | not-equal);
        value next-header-value;
    }
    payload-length {
        match (equal | greater-than | less-than | not-equal);
        value payload-length-value;
    }
    source {
        match (equal | greater-than | less-than | not-equal);
        value ip-address-or-hostname;
    }
    traffic-class {
        match (equal | greater-than | less-than | not-equal);
        value traffic-class-value;
    }
tcp {
    ack-number {
        match (equal | greater-than | less-than | not-equal);
        value acknowledgement-number;
    }
    checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
    }
    data-length {
        match (equal | greater-than | less-than | not-equal);
        value tcp-data-length;
    }
    destination-port {
        match (equal | greater-than | less-than | not-equal);
        value destination-port;
    }
    header-length {
        match (equal | greater-than | less-than | not-equal);
    value header-length;
  }
  mss {
    match (equal | greater-than | less-than | not-equal);
    value maximum-segment-size;
  }
  option {
    match (equal | greater-than | less-than | not-equal);
    value tcp-option;
  }
  reserved {
    match (equal | greater-than | less-than | not-equal);
    value reserved-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  source-port {
    match (equal | greater-than | less-than | not-equal);
    value source-port;
  }
  tcp-flags {
    (ack | no-ack);
    (fin | no-fin);
    (psh | no-psh);
    (r1 | no-r1);
    (r2 | no-r2);
    (rst | no-rst);
    (syn | no-syn);
    (urg | no-urg);
  }
  urgent-pointer {
    match (equal | greater-than | less-than | not-equal);
    value urgent-pointer;
  }
  window-scale {
    match (equal | greater-than | less-than | not-equal);
    value window-scale-factor;
  }
  window-size {
    match (equal | greater-than | less-than | not-equal);
    value window-size;
  }
}
udp {
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value data-length;
  }
  destination-port {
    match (equal | greater-than | less-than | not-equal);
```

```
      value destination-port;
    }
    source-port {
      match (equal | greater-than | less-than | not-equal);
      value source-port;
    }
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3. Statement modified in Junos OS Release 12.3X48-D25 to add ICMPv6 protocol support for custom attacks.

Description Specify a protocol to match the header information for the signature attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

protocol-binding

Supported Platforms SRX Series, vSRX

Syntax

```
protocol-binding {
   application application-name;
   icmp;
   icmpv6;
   ip {
      protocol-number transport-layer-protocol-number;
   }
   ipv6 {
      protocol-number transport-layer-protocol-number;
   }
   rpc {
      program-number rpc-program-number;
   }
   tcp {
      minimum-port port-number <maximum-port port-number>;
   }
   udp {
      minimum-port port-number <maximum-port port-number>;
   }
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type chain]
[edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Select a protocol that the attack uses to enter your network.

Options The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

protocol-name

Supported Platforms SRX Series, vSRX

Syntax

```
protocol-name protocol-name {
   tunable-name  tunable-name {
      tunable-value protocol-value;
   }
}
```

Hierarchy Level [edit security idp sensor-configuration detector]

Release Information Statement introduced in Junos OS Release 9.2. Support for file format decoding over HTTP using MIME added in Junos OS Release 11.2.

Description Specify the name of the protocol to be used to configure each of the protocol detector engines.

Options ***protocol-name***—Name of the specific protocol.

The remaining statements are explained separately. See CLI Explorer.

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

re-assembler

Supported Platforms SRX Series, vSRX

Syntax

```
re-assembler {
  action-on-reassembly-failure (drop | drop-session | ignore);
  (force-tcp-window-checks | no-force-tcp-window-checks);
  (ignore-memory-overflow | no-ignore-memory-overflow);
  (ignore-reassembly-memory-overflow | no-ignore-reassembly-memory-overflow);
  ignore-reassembly-overflow;
  max-flow-mem value;
  max-packet-mem-ratio percentage-value;
  max-synacks-queued value;
  (tcp-error-logging | no-tcp-error-logging);
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.2. Packet memory ratios added in Junos OS Release 12.1X44-D20.

Description Configure TCP reassembler for IDP sensor settings.

Options The remaining statements are explained separately. See CLI Explorer.

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

recommended

Supported Platforms SRX Series, vSRX

Syntax

```
recommended;
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify recommended filter to add predefined attacks recommended by Juniper Networks to the dynamic attack group.

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

recommended-action

Supported Platforms SRX Series, vSRX

Syntax recommended-action (close | close-client | close-server | drop | drop-packet | ignore | none);

Hierarchy Level [edit security idp custom-attack *attack-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description When the security device detects an attack, it performs the specified action.

Options The seven actions are as follows, from most to least severe:

- **close**—Reset the client and the server.
- **close-client**—Reset the client.
- **close-server**—Reset the server.
- **drop**—Drop the particular packet and all subsequent packets of the flow.
- **drop-packet**—Drop the particular packet of the flow.
- **ignore**—Do not inspect any further packets.
- **none**—Do not perform any action.

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

refresh-timeout

Supported Platforms SRX Series, vSRX

Syntax refresh-timeout;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 10.2.

Description Refresh the ip-action timeout so it does not expire when future connections match the installed ip-action filter.

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

regexp

Supported Platforms SRX Series, vSRX

Syntax regexp *regular-expression*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a Perl Compatible Regular Expression (PCRE) expression.

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

reject-timeout

Supported Platforms SRX Series, vSRX

Syntax reject-timeout *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the amount of time in seconds within which a response must be received.

This time-out is applied on flow when drop-connection action is taken by IPS for TCP flow.

Options *value*—Maximum amount of time in seconds.

Range: 1 through 65,535 seconds

Default: 300 seconds

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

reserved (Security IDP Custom Attack)

Supported Platforms SRX Series

Syntax

```
reserved {
  match (equal | greater-than | less-than | not-equal);
  value reserved-value;
}
```

Hierarchy Level [edit security idp custom-attack *ipv4_custom* attack-type signature protocol *tcp*]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the three reserved bits in the TCP header field.

Options **match (equal | greater-than | less-than | not-equal)**—Match an operand.

value—Match a decimal value.

Range: 0 through 7

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

reset (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
reset;
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain]

Release Information Statement introduced in Junos OS Release 9.3.

Description Select **reset** if the compound attack should be matched more than once within a single session or transaction.

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

reset-on-policy

Supported Platforms SRX Series, vSRX

Syntax

```
(reset-on-policy | no-reset-on-policy);
```

Hierarchy Level

```
[edit security idp sensor-configuration flow]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description IDP keeps track of connections in a table. If enabled, the security module resets the flow table each time a security policy loads or unloads. If this setting is disabled, then the security module continues to retain a previous security policy until all flows referencing that security policy go away. Juniper Networks recommends that you keep this setting enabled to preserve memory.

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

rewrite-rules (CoS Interfaces)

Supported Platforms SRX Series, vSRX

Syntax

```
rewrite-rules {
  dscp (rewrite-name | default);
  dscp-ipv6 (rewrite-name | default);
  exp (rewrite-name | default) protocol protocol-types;
  exp-push-push-push default;
  exp-swap-push-push default;
  ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
  inet-precedence (rewrite-name | default);
}
```

Hierarchy Level [edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced in Release 8.5 of Junos OS.
The option to apply IEEE 802.1 rewrite rules to both inner and outer VLAN tags introduced for SRX Series devices in Junos OS Release 18.1.

Description Associate a rewrite-rules configuration or default mapping with a specific interface.

Options

- *rewrite-name*—Name of a **rewrite-rules** mapping configured at the **[edit class-of-service rewrite-rules]** hierarchy level.
- **default**—The default mapping.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *rewrite-rules (CoS)*

routing-header

Supported Platforms SRX Series

Syntax

```
routing-header {
   header-type {
      match (equal | greater-than | less-than | not-equal);
      value header-value;
   }
}
```

Hierarchy Level [edit set security idp custom-attack *attack-name* attack-type signature protocol *ipv6* extension-header]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the IPv6 routing header type. The **routing-header** option inspects the routing-header type field and reports a custom attack if a match with the specified value is found. The **routing-header** option supports the following routing header types: **routing-header-type0**, **routing-header-type1**, and so on.

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

rpc

Supported Platforms SRX Series, vSRX

Syntax

```
rpc {
   program-number rpc-program-number;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the attack for a specified remote procedure call (RPC) program number.

Options **program-number** ***rpc-program-number***—RPC program number.

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

rule (Security Exempt Rulebase)

Supported Platforms SRX Series, vSRX

Syntax

```
rule rule-name {
   description text;
   match {
      attacks {
         custom-attack-groups [attack-group-name];
         custom-attacks [attack-name];
         dynamic-attack-groups [attack-group-name];
         predefined-attack-groups [attack-group-name];
         predefined-attacks [attack-name];
      }
      destination-address ([address-name] | any | any-ipv4 | any-ipv6);
      destination-except [address-name];
      from-zone (zone-name | any );
      source-address ([address-name] | any | any-ipv4 | any-ipv6);
      source-except [address-name];
      to-zone (zone-name | any);
   }
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify exempt rule to create, modify, delete, and reorder the rules in a rulebase.

Options ***rule-name***—Name of the exempt rulebase rule.

The remaining statements are explained separately. See CLI Explorer.

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

rule (Security IPS Rulebase)

Supported Platforms SRX Series, vSRX

Syntax

```
rule rule-name {
  description text;
  match {
    application (application-name | any | default);
    attacks {
      custom-attack-groups [attack-group-name];
      custom-attacks [attack-name];
      dynamic-attack-groups [attack-group-name];
      predefined-attack-groups [attack-group-name];
      predefined-attacks [attack-name];
    }
    destination-address ([address-name] | any | any-ipv4 | any-ipv6);
    destination-except [address-name];
    from-zone (zone-name | any );
    source-address ([address-name] | any | any-ipv4 | any-ipv6);
    source-except [address-name];
    to-zone (zone-name | any);
  }
  terminal;
  then {
    action {
      class-of-service {
        dscp-code-point number;
        forwarding-class forwarding-class;
      }
      (close-client | close-client-and-server | close-server |drop-connection | drop-packet
        | ignore-connection | mark-diffserv value | no-action | recommended);
    }
    ip-action {
      (ip-block | ip-close | ip-notify);
      log;
      log-create;
      refresh-timeout;
      target (destination-address | service | source-address | source-zone |
        source-zone-address | zone-service);
      timeout seconds;
    }
    notification {
      log-attacks {
        alert;
      }
      packet-log {
        post-attack number;
        post-attack-timeout seconds;
        pre-attack number;
      }
    }
    severity (critical | info | major | minor | warning);
  }
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify IPS rule to create, modify, delete, and reorder the rules in a rulebase.

Options ***rule-name***—Name of the IPS rulebase rule.

The remaining statements are explained separately. See CLI Explorer.

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

rulebase-exempt

Supported Platforms

SRX Series, vSRX

Syntax

```
rulebase-exempt {
  rule rule-name {
    description text;
    match {
      attacks {
        custom-attack-groups [attack-group-name];
        custom-attacks [attack-name];
        dynamic-attack-groups [attack-group-name];
        predefined-attack-groups [attack-group-name];
        predefined-attacks [attack-name];
      }
      destination-address ([address-name] | any | any-ipv4 | any-ipv6);
      destination-except [address-name];
      from-zone (zone-name | any );
      source-address ([address-name] | any | any-ipv4 | any-ipv6);
      source-except [address-name];
      to-zone (zone-name | any);
    }
  }
}
```

Hierarchy Level

[edit security idp idp-policy *policy-name*]

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Configure the exempt rulebase to skip detection of a set of attacks in certain traffic.

NOTE: You must configure the IPS rulebase before configuring the exempt rulebase.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

rulebase-ips

Supported Platforms SRX Series, vSRX

Syntax

```
rulebase-ips {
  rule rule-name {
    description text;
    match {
      application (application-name | any | default);
      attacks {
        custom-attack-groups [attack-group-name];
        custom-attacks [attack-name];
        dynamic-attack-groups [attack-group-name];
        predefined-attack-groups [attack-group-name];
        predefined-attacks [attack-name];
      }
      destination-address ([address-name] | any | any-ipv4 | any-ipv6);
      destination-except [address-name];
      from-zone (zone-name | any );
      source-address ([address-name] | any | any-ipv4 | any-ipv6);
      source-except [address-name];
      to-zone (zone-name | any);
    }
    terminal;
    then {
      action {
        class-of-service {
          dscp-code-point number;
          forwarding-class forwarding-class;
        }
        (close-client | close-client-and-server | close-server |drop-connection | drop-packet
          | ignore-connection | mark-diffserv value | no-action | recommended);
      }
      ip-action {
        (ip-block | ip-close | ip-notify);
        log;
        log-create;
        refresh-timeout;
        target (destination-address | service | source-address | source-zone |
          source-zone-address | zone-service);
        timeout seconds;
      }
      notification {
        log-attacks {
          alert;
        }
        packet-log {
          post-attack number;
          post-attack-timeout seconds;
          pre-attack number;
        }
      }
      severity (critical | info | major | minor | warning);
    }
  }
}
```

Hierarchy Level [edit security idp idp-policy *policy-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure the IPS rulebase to detect attacks based on stateful signature and protocol anomalies.

Options The remaining statements are explained separately. See CLI Explorer.

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

scope (Security IDP Chain Attack)

Supported Platforms SRX Series, vSRX

Syntax scope (session | transaction);

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify whether the match should occur over a single session or can be made across multiple transactions within a session.

Options

- **session**—Allow multiple matches for the object within the same session.
- **transaction**—Match the object across multiple transactions that occur within the same session.

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

scope (Security IDP Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax scope (destination | peer | source);

Hierarchy Level [edit security idp custom-attack *attack-name* time-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify whether the counting of the attack is from the same source IP address, the same destination IP address, or a peer.

Options

- **destination**—IDP detects attacks to a given destination IP address for the specified number of times, regardless of the source IP address.
- **peer**—IDP detects attacks between source and destination IP addresses of the sessions for the specified number of times.
- **source**—IDP detects attacks from a given source IP address for the specified number of times, regardless of the destination IP address.

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

security-package

Supported Platforms SRX Series, vSRX

Syntax

```
security-package {
  automatic {
    download-timeout minutes;
    enable;
    interval hours;
    start-time start-time;
  }
  install {
    ignore-version-check;
  }
  source-address address;
  url url-name;
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure the device to automatically download the updated signature database from the specified URL.

Options The remaining statements are explained separately. See CLI Explorer.

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

sensor-configuration

Supported Platforms SRX Series, vSRX

Syntax

```
sensor-configuration {
  application-identification {
    max-packet-memory-ratio percentage-value;
    max-reass-packet-memory-ratio percentage-value;
    max-tcp-session-packet-memory value;
    max-udp-session-packet-memory value;
  }
  detector {
    protocol-name protocol-name {
      tunable-name  tunable-name {
        tunable-value protocol-value;
      }
    }
  }
  flow {
    (allow-icmp-without-flow | no-allow-icmp-without-flow);
    fifo-max-size value;
    drop-if-no-policy-loaded;
    drop-on-failover;
    drop-on-limit;
    hash-table-size value;
    (log-errors | no-log-errors);
    max-sessions-offset value;
    max-timers-poll-ticks value;
    min-objcache-limit-lt lower-threshold-value;
    min-objcache-limit-ut upper-threshold-value;
    reject-timeout value;
    (reset-on-policy | no-reset-on-policy);
    udp-anticipated-timeout value;
  }
  global {
    (enable-all-qmodules | no-enable-all-qmodules);
    (enable-packet-pool | no-enable-packet-pool);
    memory-limit-percent value;
    (policy-lookup-cache | no-policy-lookup-cache);
  }
  high-availability {
    no-policy-cold-synchronization;
  }
  ips {
    content-decompression-max-memory-kb value;
    content-decompression-max-ratio value;
    (detect-shellcode | no-detect-shellcode);
    fifo-max-size value;
    (ignore-regular-expression | no-ignore-regular-expression);
    log-supercede-min minimum-value;
    pre-filter-shellcode;
    (process-ignore-s2c | no-process-ignore-s2c);
    (process-override | no-process-override);
    process-port port-number;
  }
  log {
    cache-size size;
    suppression {
      disable;
      (include-destination-address | no-include-destination-address);
      max-logs-operate value;
      max-time-report value;
      start-log value;
    }
  }
  packet-log {
    host ip-address <port number>;
    max-sessions percentage;
    source-address ip-address;
    total-memory percentage;
  }
  re-assembler {
    action-on-reassembly-failure (drop | drop-session | ignore);
    (force-tcp-window-checks | no-force-tcp-window-checks);
    (ignore-memory-overflow | no-ignore-memory-overflow);
    (ignore-reassembly-memory-overflow | no-ignore-reassembly-memory-overflow);
    ignore-reassembly-overflow;
    max-flow-mem value;
    max-packet-mem-ratio percentage-value;
    max-synacks-queued value;
    (tcp-error-logging | no-tcp-error-logging);
  }
  ssl-inspection {
    cache-prune-chunk-size number;
    key-protection;
    maximum-cache-size number;
    session-id-cache-timeout seconds;
    sessions number;
  }
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.2. Packet memory ratios added in Junos OS Release 12.1X44-D20.

Description Configure various IDP parameters to match the properties of transiting network traffic.

Options The remaining statements are explained separately. See CLI Explorer.

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

sequence-number (Security IDP ICMP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
sequence-number {
   match (equal | greater-than | less-than | not-equal);
   value sequence-number;
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type signature protocol icmp]
[edit security idp custom-attack *attack-name* attack-type signature protocol icmpv6]

Release Information Statement introduced in Junos OS Release 9.3. Statement modified in Junos OS Release 12.3X48-D25 to add ICMPv6 protocol support for custom attacks.

Description Specify the sequence number of the packet. This number identifies the location of the request/reply in relation to the entire sequence.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***sequence-number***—Match a decimal value.

Range: 0 through 65,535

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

sequence-number (Security IDP TCP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
sequence-number {
  match (equal | greater-than | less-than | not-equal);
  value sequence-number;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the sequence number of the packet. This number identifies the location of the data in relation to the entire data sequence.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***sequence-number***—Match a decimal value.

Range: 0 through 4,294,967,295

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

service (Security IDP Anomaly Attack)

Supported Platforms SRX Series, vSRX

Syntax service *service-name*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type anomaly]

Release Information Statement introduced in Junos OS Release 9.3.

Description Service is the protocol whose anomaly is defined in the attack. IP, TCP, UDP, and ICMP are also valid as services. (Protocol names must be entered in lowercase.)

Options *service-name*—Name of the protocol in lowercase.

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

service (Security IDP Dynamic Attack Group)

Supported Platforms SRX Series, vSRX

Syntax

```
service {
  values [service-value];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a service filter to add attack objects based on the attack service, such as FTP, HTTP, NetBios, and so on.

Options **values**—Name of the service filter. You can configure multiple filters separated by spaces and enclosed in square brackets.

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

session-id-cache-timeout

Supported Platforms SRX Series, vSRX

Syntax

```
session-id-cache-timeout seconds;
```

Hierarchy Level [edit security idp sensor-configuration ssl inspection]

Release Information Statement introduced in Junos OS Release 9.2.

Description Sets the timeout value for an IDP session ID cache (range: 1 through 7200 seconds).

Options The remaining statements are explained separately. See CLI Explorer.

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

sessions

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
sessions number;
```

Hierarchy Level

```
[edit security idp sensor-configuration ssl-inspection]
```

Release Information Statement introduced in Junos OS Release 9.3.

Description Maximum number of SSL sessions for inspection. This limit is per Services Processing Unit (SPU).

Options *number*—Number of SSL session to inspect.

Range: 1 through 100,000

Default: 10,000

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

severity (Security IDP Custom Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
severity (critical | info | major | minor | warning);
```

Hierarchy Level

```
[edit security idp custom-attack attack-name]
```

Release Information Statement introduced in Junos OS Release 9.3.

Description Select the severity that matches the lethality of the attack object on your network.

Options You can set the severity level to the following levels:

- **critical**—Contains attack objects matching exploits that attempt to evade detection, cause a network device to crash, or gain system-level privileges.
- **info**—Contains attack objects matching normal, harmless traffic containing URLs, DNS lookup failures, SNMP public community strings, and Peer-to-Peer (P2P) parameters. You can use informational attack objects to obtain information about your network.
- **major**—Contains attack objects matching exploits that attempt to disrupt a service, gain user-level access to a network device, or activate a Trojan horse previously loaded on a device.
- **minor**—Contains attack objects matching exploits that detect reconnaissance efforts attempting to access vital information through directory traversal or information leaks.
- **warning**—Contains attack objects matching exploits that attempt to obtain noncritical information or scan a network with a scanning tool.

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

severity (Security IDP Dynamic Attack Group)

Supported Platforms SRX Series, vSRX

Syntax

```
severity {
  values [critical info major minor warning];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a severity filter to add attack objects based on the attack severity levels.

Options **values**—Name of the severity filter. You can select from the following severity:

- **critical**—The attack is a critical one.
- **info**—Provide information of attack when it matches.
- **major**—The attack is a major one.
- **minor**—The attack is a minor one.
- **warning**—Issue a warning when attack matches.

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

severity (Security IDP IPS Rulebase)

Supported Platforms SRX Series, vSRX

Syntax

```
severity (critical | info | major | minor | warning);
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set the rule severity levels in logging to support better organization and presentation of log records on the log server. You can use the default severity settings of the selected attack object, or choose a specific severity for your rule. The severity you configure in the rules overrides the inherited attack severity.

Options You can set the severity level to the following levels:

- **critical**—2
- **info**—3
- **major**—4
- **minor**—5
- **warning**—7

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

shellcode

Supported Platforms SRX Series, vSRX

Syntax

```
shellcode (all | intel | no-shellcode | sparc);
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type anomaly]
[edit security idp custom-attack *attack-name* attack-type signature]

Release Information Statement introduced in Junos OS Release 9.3.

Description Shellcode signifies that the attack is a shellcode attack and is capable of creating its own shell.

Options

- **all**—All shellcode checks will be performed if this attack matches.
- **intel**—Basic shellcode checks and Intel-specific shellcode checks will be performed.
- **no-shellcode**—No shellcode checks will be performed.
- **sparc**—Basic shellcode checks and Sparc-specific shellcode checks will be performed.

Default: Basic shellcode checks will be performed when this field is not configured.

Required Privilege Level

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

signature (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
signature {
  context context-name;
  direction (any | client-to-server | server-to-client);
  negate;
  pattern signature-pattern;
  pattern-pcre signature-pattern-pcre;
  protocol {
    icmp {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      code {
        match (equal | greater-than | less-than | not-equal);
        value code-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value data-length;
      }
      identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
      }
      sequence-number {
        match (equal | greater-than | less-than | not-equal);
        value sequence-number;
      }
      type {
        match (equal | greater-than | less-than | not-equal);
        value type-value;
      }
    }
    icmpv6 {
      checksum-validate {
        match (equal | greater-than | less-than | not-equal);
        value checksum-value;
      }
      code {
        match (equal | greater-than | less-than | not-equal);
        value code-value;
      }
      data-length {
        match (equal | greater-than | less-than | not-equal);
        value data-length;
      }
      identification {
        match (equal | greater-than | less-than | not-equal);
        value identification-value;
      }
```

```
   sequence-number {
      match (equal | greater-than | less-than | not-equal);
      value sequence-number;
   }
   type {
      match (equal | greater-than | less-than | not-equal);
      value type-value;
   }
}
ipv4 {
   checksum-validate {
      match (equal | greater-than | less-than | not-equal);
      value checksum-value;
   }
   destination {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
   }
   identification {
      match (equal | greater-than | less-than | not-equal);
      value identification-value;
   }
   ihl {
      match (equal | greater-than | less-than | not-equal);
      value ihl-value;
   }
   ip-flags {
      (df | no-df);
      (mf | no-mf);
      (rb | no-rb);
   }
   protocol {
      match (equal | greater-than | less-than | not-equal);
      value transport-layer-protocol-id;
   }
   source {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
   }
   tos {
      match (equal | greater-than | less-than | not-equal);
      value type-of-service-in-decimal;
   }
   total-length {
      match (equal | greater-than | less-than | not-equal);
      value total-length-of-ip-datagram;
   }
   ttl {
      match (equal | greater-than | less-than | not-equal);
      value time-to-live;
   }
}
ipv6 {
   destination {
      match (equal | greater-than | less-than | not-equal);
      value ip-address-or-hostname;
```

```
  }
  extension-header {
    destination-option {
      home-address {
        match (equal | greater-than | less-than | not-equal);
        value header-value;
      }
        option-type {
          match (equal | greater-than | less-than | not-equal);
          value header-value;
        }
    }
    routing-header {
      header-type {
        match (equal | greater-than | less-than | not-equal);
        value header-value;
        }
      }
  }
  flow-label {
    match (equal | greater-than | less-than | not-equal);
    value flow-label-value;
  }
  hop-limit {
    match (equal | greater-than | less-than | not-equal);
    value hop-limit-value;
  }
  next-header {
    match (equal | greater-than | less-than | not-equal);
    value next-header-value;
  }
  payload-length {
    match (equal | greater-than | less-than | not-equal);
    value payload-length-value;
  }
  source {
    match (equal | greater-than | less-than | not-equal);
    value ip-address-or-hostname;
  }
  traffic-class {
    match (equal | greater-than | less-than | not-equal);
    value traffic-class-value;
  }
tcp {
  ack-number {
    match (equal | greater-than | less-than | not-equal);
    value acknowledgement-number;
  }
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value tcp-data-length;
  }
  destination-port {
    match (equal | greater-than | less-than | not-equal);
    value destination-port;
  }
  header-length {
    match (equal | greater-than | less-than | not-equal);
    value header-length;
  }
  mss {
    match (equal | greater-than | less-than | not-equal);
    value maximum-segment-size;
  }
  option {
    match (equal | greater-than | less-than | not-equal);
    value tcp-option;
  }
  reserved {
    match (equal | greater-than | less-than | not-equal);
    value reserved-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  source-port {
    match (equal | greater-than | less-than | not-equal);
    value source-port;
  }
  tcp-flags {
    (ack | no-ack);
    (fin | no-fin);
    (psh | no-psh);
    (r1 | no-r1);
    (r2 | no-r2);
    (rst | no-rst);
    (syn | no-syn);
    (urg | no-urg);
  }
  urgent-pointer {
    match (equal | greater-than | less-than | not-equal);
    value urgent-pointer;
  }
  window-scale {
    match (equal | greater-than | less-than | not-equal);
    value window-scale-factor;
  }
  window-size {
    match (equal | greater-than | less-than | not-equal);
    value window-size;
  }
}
udp {
  checksum-validate {
    match (equal | greater-than | less-than | not-equal);
    value checksum-value;
  }
```

```
            data-length {
              match (equal | greater-than | less-than | not-equal);
              value data-length;
            }
            destination-port {
              match (equal | greater-than | less-than | not-equal);
              value destination-port;
            }
            source-port {
              match (equal | greater-than | less-than | not-equal);
              value source-port;
            }
          }
        }
        protocol-binding {
          application application-name;
          icmp;
          icmpv6;
          ip {
            protocol-number transport-layer-protocol-number;
          }
          ipv6 {
            protocol-number transport-layer-protocol-number;
          }
          rpc {
            program-number rpc-program-number;
          }
          tcp {
            minimum-port port-number <maximum-port port-number>;
          }
          udp {
            minimum-port port-number <maximum-port port-number>;
          }
        }
        regexp regular-expression;
        shellcode (all | intel | no-shellcode | sparc);
      }
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type]

Release Information Statement introduced in Junos OS Release 9.3.

Description IDP uses stateful signatures to detect attacks. Stateful signatures are more specific than regular signatures. With stateful signatures, IDP can look for the specific protocol or service used to perpetrate the attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

source (Security IDP IP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
source {
   match (equal | greater-than | less-than | not-equal);
   value ip-address-or-hostname;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the IP address or hostname of the attacking device.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***ip-address-or-hostname***—Match an IP address or a hostname.

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

source-address (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
source-address address;
```

Hierarchy Level [edit security idp security-package]

Description Sets the source address to be used for sending download requests.

Options The remaining statements are explained separately. See CLI Explorer.

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

source-address (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax source-address ([*address-name*] | any | any-ipv4 | any-ipv6);

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a source IP address or IP address set object to be used as the match source address object. The default value is any.

Options

- ***address-name***—IP address or IP address set object.
- ***any***—Specify any IPv4 or IPv6 address.
- ***any-ipv4***—Specify any IPv4 address.
- ***any-ipv6***—Specify any IPv6 address.

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

source-address (Security IDP Sensor Configuration)

Supported Platforms SRX1500, SRX5400, SRX5600, SRX5800, vSRX

Syntax source-address *ip-address*;

Hierarchy Level [edit security idp sensor-configuration packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the source IP address for the carrier UDP packet.

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

source-except

Supported Platforms SRX Series, vSRX

Syntax

```
source-except [address-name];
```

Hierarchy Level

```
[edit security idp idp-policy policy-name rulebase-exempt rule rule-name match]
[edit security idp idp-policy policy-name rulebase-ips rule rule-name match]
```

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a source IP address or IP address set object to specify all source address objects except the specified address objects. The default value is any.

Options ***address-name***—IP address or IP address set object.

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

source-port (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
source-port {
  match (equal | greater-than | less-than | not-equal);
  value source-port;
}
```

Hierarchy Level

```
[edit security idp custom-attack attack-name attack-type signature protocol udp]
[edit security idp custom-attack attack-name attack-type signature protocol tcp]
```

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the port number on the attacking device.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***source-port***—Port number on the attacking device.

Range: 0 through 65,535

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

ssl-inspection

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax

```
ssl-inspection {
  cache-prune-chunk-size number;
  key-protection;
  maximum-cache-size number;
  session-id-cache-timeout seconds;
  sessions number;
}
```

Hierarchy Level [edit security idp sensor-configuration]

Release Information Statement introduced in Junos OS Release 9.3.

Description Inspect HTTP traffic encrypted in SSL protocol. SSL inspection is disabled by default. It is enabled if you configure SSL inspection.

Options The remaining statements are explained separately. See CLI Explorer.

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

start-log

Supported Platforms SRX Series, vSRX

Syntax start-log *value*;

Hierarchy Level [edit security idp sensor-configuration log suppression]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify how many instances of a specific event must occur before log suppression begins.

Options *value*—Log suppression begins after how many occurrences.

Range: 1 through 128

Default: 1

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

start-time (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
start-time start-time;
```

Hierarchy Level [edit security idp security-package automatic]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the time that the device automatically starts downloading the updated signature database from the specified URL.

Options ***start-time***—Time in MM-DD.hh:mm format.

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

suppression

Supported Platforms SRX Series, vSRX

Syntax

```
suppression {
  disable;
  (include-destination-address | no-include-destination-address);
  max-logs-operate value;
  max-time-report value;
  start-log value;
}
```

Hierarchy Level [edit security idp sensor-configuration log]

Release Information Statement introduced in Junos OS Release 9.2.

Description Log suppression reduces the number of logs by displaying a single record for multiple occurrences of the same event. Log suppression can negatively impact sensor performance if the reporting interval is set too high. By default this feature is enabled.

Options **disable**—Disable log suppression.

The remaining statements are explained separately. See CLI Explorer.

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

target (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
target (destination-address | service | source-address | source-zone | source-zone-address
    | zone-service);
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the blocking options that you want to set to block the future connections. Blocking options can be based on the following matches of the attack traffic:

Options

- **destination-address**—Matches traffic based on the destination address of the attack traffic.
- **service**—For TCP and UDP, matches traffic based on the source address, source port, destination address, and destination port of the attack traffic. This is the default.

  For ICMP flows, the destination port is 0. Any ICMP flow matching source port, source address, and destination address is blocked.
- **source-address**—Matches traffic based on the source address of the attack traffic.
- **source-zone**—Matches traffic based on the source zone of the attack traffic.
- **source-zone-address**—Matches traffic based on the source zone and source address of the attack traffic.
- **zone-service**—Matches traffic based on the source zone, destination address, destination port, and protocol of the attack traffic.

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

tcp (Security IDP Protocol Binding)

Supported Platforms SRX Series, vSRX

Syntax

```
tcp {
  minimum-port port-number <maximum-port port-number>;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.2.

Description Allow IDP to match the attack for specified TCP ports.

Options **minimum-port** ***port-number***—Minimum port in the port range.

Range: 0 through 65,535

maximum-port ***port-number***—Maximum port in the port range.

Range: 0 through 65,535

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

tcp (Security IDP Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
tcp {
  ack-number {
    match (equal | greater-than | less-than | not-equal);
    value acknowledgement-number;
  }
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value tcp-data-length;
  }
  destination-port {
    match (equal | greater-than | less-than | not-equal);
    value destination-port;
  }
  header-length {
    match (equal | greater-than | less-than | not-equal);
    value header-length;
  }
  mss {
    match (equal | greater-than | less-than | not-equal);
    value maximum-segment-size;
  }
  option {
    match (equal | greater-than | less-than | not-equal);
    value tcp-option;
  }
  reserved {
    match (equal | greater-than | less-than | not-equal);
    value reserved-value;
  }
  sequence-number {
    match (equal | greater-than | less-than | not-equal);
    value sequence-number;
  }
  source-port {
    match (equal | greater-than | less-than | not-equal);
    value source-port;
  }
  tcp-flags {
    (ack | no-ack);
    (fin | no-fin);
    (psh | no-psh);
    (r1 | no-r1);
    (r2 | no-r2);
    (rst | no-rst);
    (syn | no-syn);
    (urg | no-urg);
  }
  urgent-pointer {
    match (equal | greater-than | less-than | not-equal);
    value urgent-pointer;
  }
  window-scale {
    match (equal | greater-than | less-than | not-equal);
    value window-scale-factor;
  }
  window-size {
    match (equal | greater-than | less-than | not-equal);
    value window-size;
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the TCP header information for the signature attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

tcp-flags

Supported Platforms SRX Series, vSRX

Syntax

```
tcp-flags {
  (ack | no-ack);
  (fin | no-fin);
  (psh | no-psh);
  (r1 | no-r1);
  (r2 | no-r2);
  (rst | no-rst);
  (syn | no-syn);
  (urg | no-urg);
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify that IDP looks for a pattern match whether or not the TCP flag is set.

Options

- **ack** | **no-ack**—When set, the acknowledgment flag acknowledges receipt of a packet.
- **fin** | **no-fin**—When set, the final flag indicates that the packet transfer is complete and the connection can be closed.
- **psh** | **no-psh**—When set, the push flag indicates that the receiver should push all data in the current sequence to the destination application (identified by the port number) without waiting for the remaining packets in the sequence.
- **r1** | **no-r1**—When set, indicates that the R1 retransmission threshold has been reached.
- **r2** | **no-r2**—When set, indicates that the R2 retransmission threshold has been reached.
- **rst** | **no-rst**—When set, the reset flag resets the TCP connection, discarding all packets in an existing sequence.
- **syn** | **no-syn**—When set, indicates that the sending device is asking for a three-way handshake to initialize communications.
- **urg** | **no-urg**—When set, the urgent flag indicates that the packet data is urgent.

Required Privilege Level

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

terminal

Supported Platforms SRX Series, vSRX

Syntax terminal;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set or unset a terminal rule flag. The device stops matching rules for a session when a terminal rule is matched.

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

test (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax test *test-condition*;

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type anomaly]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify protocol anomaly condition to be checked.

Options ***test-condition***—Name of the anomaly test condition.

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

then (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax

```
then {
  action {
    class-of-service {
      dscp-code-point number;
      forwarding-class forwarding-class;
    }
    (close-client | close-client-and-server | close-server |drop-connection | drop-packet |
      ignore-connection | mark-diffserv value | no-action | recommended);
  }
  ip-action {
    (ip-block | ip-close | ip-notify);
    log;
    log-create;
    refresh-timeout;
    target (destination-address | service | source-address | source-zone | source-zone-address
      | zone-service);
    timeout seconds;
  }
  notification {
    log-attacks {
      alert;
    }
    packet-log {
      post-attack number;
      post-attack-timeout seconds;
      pre-attack number;
    }
  }
  severity (critical | info | major | minor | warning);
}
```

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the action to be performed when traffic matches the defined criteria.

Options The remaining statements are explained separately. See CLI Explorer.

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

then (Security Policies)

Supported Platforms SRX Series, vSRX

Syntax

```
then {
  count {
    alarm {
      per-minute-threshold number;
      per-second-threshold number;
    }
  }
  deny;
  log {
    session-close;
    session-init;
  }
  permit {
    application-services {
      application-firewall {
        rule-set rule-set-name;
      }
      application-traffic-control {
        rule-set rule-set-name;
      }
      gprs-gtp-profile profile-name;
      gprs-sctp-profile profile-name;
      idp;
      redirect-wx | reverse-redirect-wx;
      ssl-proxy {
        profile-name profile-name;
      }
      uac-policy {
        captive-portal captive-portal;
      }
      utm-policy policy-name;
    }
    destination-address {
      drop-translated;
      drop-untranslated;
    }
    firewall-authentication {
      pass-through {
        access-profile profile-name;
        client-match user-or-group-name;
        ssl-termination-profile profile-name;
        web-redirect;
        web-redirect-to-https;
      }
      user-firewall {
        access-profile profile-name;
        domain domain-name
        ssl-termination-profile profile-name;
      }
      web-authentication {
        client-match user-or-group-name;
      }
    }
    services-offload;
    tcp-options {
      initial-tcp-mss mss-value;
      reverse-tcp-mss mss-value;
      sequence-check-required;
      syn-check-required;
    }
    tunnel {
      ipsec-group-vpn group-vpn;
      ipsec-vpn vpn-name;
      pair-policy pair-policy;
    }
  }
  reject;
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name*]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **services-offload** option added in Junos OS Release 11.4. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20.

Description Specify the policy action to be performed when packets match the defined criteria.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

time-binding

Supported Platforms SRX Series, vSRX

Syntax

```
time-binding {
  count count-value;
  scope (destination | peer | source);
}
```

Hierarchy Level [edit security idp custom-attack *attack-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to detect a sequence of the same attacks over a period of time.

Options The remaining statements are explained separately. See CLI Explorer.

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

timeout (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax timeout *seconds*;

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* then ip-action]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the number of seconds that you want the IP action to remain in effect after a traffic match.

Options *seconds*—Number of seconds the IP action should remain effective.

Range: 0 through 64,800 seconds

Default: 0 second

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

tos

Supported Platforms SRX Series, vSRX

Syntax

```
tos {
  match (equal | greater-than | less-than | not-equal);
  value type-of-service-in-decimal;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the type of service.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***type-of-service-in-decimal***—The following service types are available:
  - 0000—Default
  - 0001—Minimize Cost
  - 0002—Maximize Reliability
  - 0003—Maximize Throughput
  - 0004—Minimize Delay
  - 0005—Maximize Security

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

total-length

Supported Platforms SRX Series, vSRX

Syntax

```
total-length {
  match (equal | greater-than | less-than | not-equal);
  value total-length-of-ip-datagram;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the number of bytes in the packet, including all header fields and the data payload.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***total-length-of-ip-datagram***—Length of the IP datagram.

Range: 0 through 65,535

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

total-memory

Supported Platforms SRX1500, SRX5400, SRX5600, SRX5800, vSRX

Syntax total-memory *percentage*;

Hierarchy Level [edit security idp sensor-configuration packet-log]

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the maximum amount of memory to be allocated to packet capture for the device. This value is expressed as a percentage of the memory available on the device. The total memory for a device will differ depending on its operating mode.

Options

- *percentage*—Amount of packet capture memory expressed as a percentage of total memory for the device mode.

Range: 1 to 100 percent

Default: 10

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

to-zone (Security IDP Policy)

Supported Platforms SRX Series, vSRX

Syntax to-zone (*zone-name* | any);

Hierarchy Level [edit security idp idp-policy *policy-name* rulebase-exempt rule *rule-name* match]
[edit security idp idp-policy *policy-name* rulebase-ips rule *rule-name* match]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify a destination zone to be associated with the security policy. The default value is any.

Options ***zone-name***—Name of the destination zone object.

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

traceoptions (Security Datapath Debug)

Supported Platforms SRX Series

Syntax

```
traceoptions {
  file {
    filename;
    files number;
    match regular-expression;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  no-remote-trace;
}
```

Hierarchy Level [edit security datapath-debug]

Release Information Command introduced in Junos OS Release 9.6.

Description Sets the trace options for datapath-debug.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Options

- **file**—Configure the trace file options.
  - **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**. By default, the name of the file is the name of the process being traced.
  - **files** ***number***—Maximum number of trace files. When a trace file named **trace-file** reaches its maximum size, it is renamed to **trace-file.0**, then **trace-file.1**, and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten.

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

    Range: 2 through 1000 files

    Default: 10 files
  - **match** ***regular-expression***—Refine the output to include lines that contain the regular expression.
  - **size** ***maximum-file-size***—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.

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

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

Range: 10 KB through 1 GB

Default: 128 KB

  - **world-readable | no-world-readable**—By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option

- **no-remote-trace**—Set remote tracing as disabled.

Required Privilege Level

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

traceoptions (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
traceoptions {
  file {
    filename;
    files number;
    match regular-expression;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  flag all;
  level (all | error | info | notice | verbose | warning);
  no-remote-trace;
}
```

Hierarchy Level [edit security idp]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure IDP tracing options.

Options

- **file**—Configure the trace file options.
  - ***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**. By default, the name of the file is the name of the process being traced.
  - **files *number***—Maximum number of trace files. When a trace file named **trace-file** reaches its maximum size, it is renamed to **trace-file.0** then **trace-file.1** and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten.

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

    Range: 2 through 1000 files

    Default: 10 files
  - **match *regular-expression***—Refine the output to include lines that contain the regular expression.
  - **size *maximum-file-size***—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 **trace-file.0** again reaches its maximum size, **trace-file.1** is renamed **trace-file.2** and **trace-file.0** is renamed **trace-file.1**. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

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

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

Range: 10 KB through 1 GB

Default: 128 KB

  - **world-readable** | **no-world-readable**—By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

- **flag**—Trace operation to perform.
  - **all**—Trace with all flags enabled
- **level**—Set the level of debugging the output option.
  - **all**—Match all levels
  - **error**—Match error conditions
  - **info**—Match informational messages
  - **notice**—Match conditions that should be handled specially
  - **verbose**—Match verbose messages
  - **warning**—Match warning messages
- **no-remote-trace**—Set remote tracing as disabled.

Required Privilege Level

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

ttl (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax

```
ttl {
  match (equal | greater-than | less-than | not-equal);
  value time-to-live;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol ipv4]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the time-to-live (TTL) value of the packet. This value represents the number of routers the packet can pass through. Each router that processes the packet decrements the TTL by 1; when the TTL reaches 0, the packet is discarded.

Options **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.

value ***time-to-live***—The time-to-live value.

Range: 0 through 255

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

tunable-name

Supported Platforms SRX Series, vSRX

Syntax

```
tunable-name  tunable-name {
   tunable-value protocol-value;
}
```

Hierarchy Level [edit security idp sensor-configuration detector protocol-name *protocol-name*]

Release Information Statement introduced in Junos OS Release 9.2. Support for file format decoding over HTTP using MIME added in Junos OS Release 11.2.

Description Specify the name of the tunable parameter to enable or disable the protocol detector for each of the service. By default, the protocol decoders for all services are enabled.

Options ***tunable-name***—Name of the specific tunable parameter.

The remaining statements are explained separately. See CLI Explorer.

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

tunable-value

Supported Platforms SRX Series, vSRX

Syntax

```
tunable-value protocol-value;
```

Hierarchy Level [edit security idp sensor-configuration detector protocol-name *protocol-name* tunable-name *tunable-name*]

Release Information Statement introduced in Junos OS Release 9.2. Support for file format decoding over HTTP using MIME added in Junos OS Release 11.2.

Description Specify the value of the tunable parameter to enable or disable the protocol detector for each of the services.

Options ***tunable-value***—Integer representing a selected option for the switch specified in **tunable-name**. The range of values depends on the options defined for the specified switch.

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

type (Security IDP Dynamic Attack Group)

Supported Platforms SRX Series, vSRX

Syntax

```
type {
  values [anomaly signature];
}
```

Hierarchy Level [edit security idp dynamic-attack-group *dynamic-attack-group-name* filters]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify an attack type filter to add attack objects based on the type of attack object (signature or protocol anomaly).

Options **values**—Name of the attack type filter.

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

type (Security IDP ICMP Headers)

Supported Platforms SRX Series, vSRX

Syntax

```
type {
  match (equal | greater-than | less-than | not-equal);
  value type-value;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol icmp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the primary code that identifies the function of the request/reply.

Options **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.

value ***type-value***—Match a decimal value.

Range: 0 through 255

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

udp (Security IDP Protocol Binding)

Supported Platforms SRX Series, vSRX

Syntax

```
udp {
   minimum-port port-number <maximum-port port-number>;
}
```

Hierarchy Level

[edit security idp custom-attack *attack-name* attack-type chain protocol-binding]
[edit security idp custom-attack *attack-name* attack-type signature protocol-binding]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the attack for specified UDP ports.

Options

- **minimum-port** ***port-number***—Minimum port in the port range.

Range: 0 through 65,535

- **maximum-port** ***port-number***—Maximum port in the port range.

Range: 0 through 65,535

Required Privilege Level

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

udp (Security IDP Signature Attack)

Supported Platforms SRX Series, vSRX

Syntax

```
udp {
  data-length {
    match (equal | greater-than | less-than | not-equal);
    value data-length;
  }
  destination-port {
    match (equal | greater-than | less-than | not-equal);
    value destination-port;
  }
  source-port {
    match (equal | greater-than | less-than | not-equal);
    value source-port;
  }
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol]

Release Information Statement introduced in Junos OS Release 9.3.

Description Allow IDP to match the UDP header information for the signature attack.

Options The remaining statements are explained separately. See CLI Explorer.

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

udp-anticipated-timeout (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax udp-anticipated-timeout *value*;

Hierarchy Level [edit security idp sensor-configuration flow]

Release Information Statement introduced in Junos OS Release 9.2.

Description Sets the maximum UDP anticipated timeout value (range: 1 through 65535).

Options The remaining statements are explained separately. See CLI Explorer.

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

urgent-pointer

Supported Platforms SRX Series, vSRX

Syntax

```
urgent-pointer {
  match (equal | greater-than | less-than | not-equal);
  value urgent-pointer;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the data in the packet is urgent; the URG flag must be set to activate this field.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***urgent-pointer***—Match the value of the urgent pointer.

Range: 0 through 65,535

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

url (Security IDP)

Supported Platforms SRX Series, vSRX

Syntax url *url-name*;

Hierarchy Level [edit security idp security-package]

Release Information Statement introduced in Junos OS Release 9.2.

Description Specify the URL to automatically download the updated signature database.

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

weight (Security)

Supported Platforms SRX Series, vSRX

Syntax weight (equal | firewall | idp);

Hierarchy Level [edit security forwarding-process application-services maximize-idp-sessions]

Release Information Statement introduced in Junos OS Release 9.6.

Description If you are deploying IDP policies, you can tune the device to increase IDP session capacity. By using the provided commands to change the way the system allocates resources, you can achieve a higher IDP session capacity.

Devices ship with an implicit default session capacity setting. This default value gives more weight to firewall sessions. You can manually override the default by using the **maximize-idp-sessions** command. The command allows you to choose between these weight values: **equal**, **firewall**, and **idp**. The following table displays the available session capacity weight and approximate throughput for each.

Table 96: Session Capacity and Resulting Throughput

Weight Value	Firewall Capacity	IDP Capacity	Firewall Throughput	IDP Throughput
Default	1,000,000	256,000	10 Gbps	2.4 Gbps
equal	1,000,000	1,000,000	8.5 Gbps	2 Gbps
firewall	1,000,000	1,000,000	10 Gbps	2.4 Gbps
idp	1,000,000	1,000,000	5.5 Gbps	1.4 Gbps

This statement is supported on SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

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

Related Documentation

- *Juniper Networks Devices Processing Overview*

window-scale

Supported Platforms SRX Series, vSRX

Syntax

```
window-scale {
   match (equal | greater-than | less-than | not-equal);
   value window-scale-factor;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the scale factor that the session of the attack will use. The window scale extension expands the definition of the TCP window to 32 bits and then uses a scale factor to carry this 32-bit value in the 16-bit window field of the TCP header.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***window-scale-factor***—Match the number of bytes.

Range: 0 through 255

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

window-size

Supported Platforms SRX Series, vSRX

Syntax

```
window-size {
   match (equal | greater-than | less-than | not-equal);
   value window-size;
}
```

Hierarchy Level [edit security idp custom-attack *attack-name* attack-type signature protocol tcp]

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the number of bytes in the TCP window size.

Options

- **match** (**equal** | **greater-than** | **less-than** | **not-equal**)—Match an operand.
- **value** ***window-size***—Match the number of bytes.

Range: 0 through 65,535

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

CHAPTER 8

Operational Commands

- show security idp counters application-identification
- show security idp counters dfa
- show security idp counters flow
- show security idp counters http-decoder
- show security idp counters ips
- show security idp counters log
- show security idp counters packet
- show security idp counters packet-log
- show security idp counters policy-manager
- show security idp counters tcp-reassembler
- show security idp logical-system policy-association
- show security idp memory
- show security idp policies
- show security idp policy-commit-status
- show security idp policy-commit-status clear
- show security idp policy-templates
- show security idp predefined-attacks
- show security idp security-package-version
- show security idp ssl-inspection key
- show security idp ssl-inspection session-id-cache
- show security idp status
- show security idp status detail

clear security datapath-debug counters

Supported Platforms SRX Series

Syntax

```
clear security datapath-debug counters
```

Release Information Command introduced in Junos OS Release 10.0.

Description Clear all data path-debugging counters.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Required Privilege Level clear

Related Documentation

- *show security datapath-debug capture*
- *show security datapath-debug counter*

Output Fields This command produces no output.

clear security idp

Supported Platforms SRX Series, vSRX

Syntax clear security idp
(application-identification | application-statistics | attack | counters | status)

Release Information Command introduced in Junos OS Release 10.1.

Description Clear the following IDP information:

- **application-identification**—Clear IDP application identification data.
- **application-statistics**—Clear IDP application statistics.
- **attack**—Clear IDP attack data
- **counters**—Clear IDP counters
- **status**—Clear IDP Status

Required Privilege Level clear

List of Sample Output clear security idp status on page 514

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

Sample Output

clear security idp status

```
user@host> clear security idp status
State of IDP: 2-default, Up since: 2010-02-04 13:37:16 UTC (17:13:45 ago)

Packets/second: 0 Peak: 0 @ 2010-02-05 06:49:51 UTC
KBits/second: 0 Peak: 0 @ 2010-02-05 06:49:51 UTC
Latency (microseconds): [min: 0] [max: 0] [avg: 0]

Packet Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]
  TCP: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]
  UDP: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]
  Other: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]

Session Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]
  Policy Name: sample
  Running Detector Version: 10.4.160091104
```

clear security idp attack table

Supported Platforms SRX Series, vSRX

Syntax clear security idp attack table

Release Information Command introduced in Junos OS Release 9.2.

Description Clear details of the IDP attack table.

Required Privilege Level clear

Related Documentation

- show security idp attack table on page 551

Output Fields This command produces no output.

clear security idp counters application-identification

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters application-identification
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the application identification counter values.

Required Privilege Level clear

Related Documentation

- application-identification on page 326
- show security idp counters application-identification on page 552

Output Fields This command produces no output.

clear security idp counters dfa

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters dfa
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the DFA counter values.

Required Privilege Level clear

Related Documentation

- show security idp counters dfa on page 556

Output Fields This command produces no output.

clear security idp counters flow

Supported Platforms	SRX Series, vSRX
Syntax	clear security idp counters flow
Release Information	Command introduced in Junos OS Release 9.2.
Description	Reset all the IDP flow-related counter values.
Required Privilege Level	clear
Related Documentation	• flow (Security IDP) on page 378 • show security idp counters flow on page 557
Output Fields	This command produces no output.

clear security idp counters http-decoder

Supported Platforms SRX Series, vSRX

Syntax clear security idp counters http-decoder

Release Information Command introduced in Junos OS Release 11.2.

Description Reset all the HTTP decoder counter values.

Required Privilege Level clear

Related Documentation

- show security idp counters http-decoder on page 564

Output Fields This command produces no output.

clear security idp counters ips

Supported Platforms	SRX Series, vSRX
Syntax	clear security idp counters ips
Release Information	Command introduced in Junos OS Release 9.2.
Description	Reset all the ips counter values.
Required Privilege Level	clear
Related Documentation	• ips on page 407 • show security idp counters ips on page 566
Output Fields	This command produces no output.

clear security idp counters log

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters log
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the IDP log counter values.

Required Privilege Level clear

Related Documentation

- *event-rate*
- show security idp counters log on page 569

Output Fields This command produces no output.

clear security idp counters packet

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters packet
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the IDP packet counter values.

Required Privilege Level clear

Related Documentation

- show security idp counters packet on page 572

Output Fields This command produces no output.

clear security idp counters policy-manager

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters policy-manager
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the IDP policies counter values.

Required Privilege Level clear

Related Documentation

- show security idp counters policy-manager on page 577

Output Fields This command produces no output.

clear security idp counters tcp-reassembler

Supported Platforms SRX Series, vSRX

Syntax

```
clear security idp counters tcp-reassembler
```

Release Information Command introduced in Junos OS Release 9.2.

Description Reset all the TCP reassembler counter values.

Required Privilege Level clear

Related Documentation

- re-assembler on page 450
- show security idp counters tcp-reassembler on page 578

Output Fields This command produces no output.

clear security idp ssl-inspection session-id-cache

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax clear security idp ssl-inspection session-id-cache

Release Information Command introduced in Junos OS Release 9.3.

Description Clear all the entries stored in the SSL session ID cache.

Required Privilege Level clear

Related Documentation

- show security idp ssl-inspection session-id-cache on page 593

List of Sample Output clear security idp ssl-inspection session-id-cache on page 525

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

Sample Output

clear security idp ssl-inspection session-id-cache

```
user@host> clear security idp ssl-inspection session-id-cache
Total SSL session cache entries cleared : 2
```

request security datapath-debug capture start

Supported Platforms SRX Series

Syntax request security datapath-debug capture start

Release Information Command introduced in Junos OS Release 10.0.

Description Start the data path debugging capture.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Required Privilege Level maintenance

Related Documentation

- *Understanding Data Path Debugging for Logical Systems*

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

Sample Output

request security datapath-debug capture start

```
user@host> request security datapath-debug capture start
datapath-debug capture started on file
```

request security idp security-package download

Supported Platforms SRX Series, vSRX

Syntax

```
request security idp security-package download
<check-server>
<full-update>
<policy-templates>
<version  version-number >
<status>
```

Release Information Command introduced in Junos OS Release 9.2. Detailed status added in Junos OS Release 10.1. Description modified in Junos OS Release 11.1. Application package support added in Junos OS Release 11.4.

Description Manually download the individual components of the security package from the Juniper Security Engineering portal. The components are downloaded into a staging folder inside the device.

By default, this command tries to download the delta set attack signature table. It also downloads IDP, IPS, and application package signatures.

Options

- **check-server**—(Optional) Retrieve the version information of the latest security package from the security portal server.
- **full-update**—(Optional) Download the latest security package with the full set of attack signature tables from the portal.
- **policy-templates**—(Optional) Download the latest policy templates from the portal.
- **version** *version-number* —(Optional) Download the security package of a specific version from the portal.
- **status**—(Optional) Provide detailed status of security package download operation.

Additional Information The **request security idp security-package download** command does not download security package files if the installed version on the device is same as the security package version on the server (https://services.netscreen.com/cgi-bin/index.cgi always). The **request security idp security-package download full-update** command downloads the latest security package files on the device from the server, irrespective of the version on the device and the server.

Required Privilege Level maintenance

Related Documentation

- show security idp active-policy on page 546
- show security idp security-package-version on page 590

List of Sample Output

request security idp security-package download on page 528
request security idp security-package download policy-templates on page 528
request security idp security-package download version 1151 full-update on page 528
request security idp security-package download status on page 528

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

Sample Output

request security idp security-package download

```
user@host> request security idp security-package download
Successfully downloaded from(https://services.netscreen.com/cgi-bin/index.cgi).
Version info:1152(Thu Apr 24 14:37:44 2008, Detector=9.1.140080400)
```

Sample Output

request security idp security-package download policy-templates

```
user@host> request security idp security-package download policy-templates
Successfully downloaded from(https://services.netscreen.com/cgi-bin/index.cgi).
Version info:35
```

Sample Output

request security idp security-package download version 1151 full-update

```
user@host> request security idp security-package download version 1151 full-update
Successfully downloaded from(https://services.netscreen.com/cgi-bin/index.cgi).
Version info:1151(Wed Apr 23 14:39:15 2008, Detector=9.1.140080400)
```

request security idp security-package download status

To request status for a package download:

```
user@host> request security idp security-package download status
Done;Successfully downloaded
from(https://services.netscreen.com/cgi-bin/index.cgi).
Version info:2014(Thu Oct 20 12:07:01 2011, Detector=11.6.140110920)
```

To request status for a template download:

```
user@host> request security idp security-package download status
Done; Successfully downloaded from
(https://services.netscreen.com/cgi-bin/index.cgi).
```

When devices are operating in chassis cluster mode, when you check the security package download status, a message is displayed confirming that the downloaded security package is being synchronized to the primary and secondary nodes.

```
user@host> request security idp security-package download status
node0:
--------------------------------------------------------------------------
Done;Successfully downloaded from(https://services.netscreen.com/cgi-bin/index.cgi)
and synchronized to backup.
Version info:2011(Mon Oct 17 15:13:06 2011, Detector=11.6.140110920)
```

request security idp security-package install

Supported Platforms SRX Series, vSRX

Syntax

```
request security idp security-package install
<policy-templates>
<status>
<update-attack-database-only>
```

Release Information Command introduced in Junos OS Release 9.2. Description modified in Junos OS Release 11.1. Added application package support in Junos OS Release 11.4.

Description Updates the attack database inside the device with the newly downloaded one from the staging folder, recompiles the existing running policy, and pushes the recompiled policy to the data plane.

Also, if there is an existing running policy, and the previously installed detector's version is different from the newly downloaded one, the downloaded components are pushed to the data plane. This command installs IDP, IPS, and application package signatures.

Options

- **policy-templates**—(Optional) Installs the policy template file into /var/db/scripts/commit/templates.
- **status**—(Optional) The command **security-package install** may take a long time depending on the new Security database size. Hence, **security-package install** command returns immediately and a background process performs the task. User can check the status using **security-package install status** command.
- **update-attack-database-only**—(Optional) Loads the security package into IDP database but does not compile/push the active policy or the new detector to the data plane.

Required Privilege Level maintenance

Related Documentation

- show security idp active-policy on page 546
- show security idp security-package-version on page 590

List of Sample Output request security idp security-package install on page 530
request security idp security-package install status on page 531

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

Sample Output

request security idp security-package install

```
user@host> request security idp security-package install
Will be processed in async mode. Check the status using the status checking CLI
```

Sample Output

request security idp security-package install status

To request status on a package installation:

```
user@host> request security idp security-package install status
Done;Attack DB update : successful - [UpdateNumber=1152,ExportDate=Thu Apr 24
14:37:44 2008]
     Updating data-plane with new attack or detector : not performed
     due to no existing active policy found.
```

To request status on a template installation:

```
user@host> request security idp security-package install status
Done; policy-template has been successfully updated into internal repository
(=>/var/db/scripts/commit/templates.xsl)!
```

request security idp security-package offline-download

Supported Platforms SRX Series

Syntax request security idp security-package offline-download (package-path *package-path* | status)

Release Information Command introduced in Junos OS Release 12.3X48-D10.

Description Unzip the security package and copy the xml files.

Manually download the security package from the Juniper Security Engineering portal. The package will have both IDP and application package signatures. Copy the files over to the device into a certain folder and then issues the **request security idp security-package offline-download package-path** ***package-path*** command. The command will unzip the security package and copy the xml files to staging directory. Signature package installation should follow an offline-download. There is no change in installation process.

Options

- **package-path**—Package path of the zipped security package.
- **status**—Retrieve the status of offline package download operation.

Required Privilege Level maintenance

Related Documentation

- show security idp active-policy on page 546
- show security idp security-package-version on page 590
- request security idp security-package install on page 530

request security idp ssl-inspection key add

Supported Platforms SRX Series, vSRX

Syntax

```
request security idp ssl-inspection key add <key-name> [file <file-name>] [password
   <password-string>] [server <server-ip>]
```

Release Information Command introduced in Junos OS Release 9.3.

Description Install a Privacy-Enhanced Mail (PEM) key that is optionally password protection, and associate a server with an installed key. The length of each key name and password string should not exceed 32 alphanumeric characters.

Options

- *key-name*—Name of the SSL private key.
- **file** *<file-name>*—(Optional) Location of RSA private key (PEM format) file.
- **password** *<password-string>*—(Optional) Password used to encrypt specified key.
- **server** *<server-ip>* —(Optional) Server IP address to be added to the specified key.

Required Privilege Level maintenance

Related Documentation

- show security idp ssl-inspection key on page 591

List of Sample Output

request security idp ssl-inspection key add key1 file /var/tmp/enc1.key password encrypted on page 533
request security idp ssl-inspection key add key2 file /var/tmp/enc2.key password encrypted on page 534
request security idp ssl-inspection key add key3 file /var/tmp/norm.key on page 534
request security idp ssl-inspection key add key1 server 1.1.0.1 on page 534
request security idp ssl-inspection key add key1 server 1.1.0.2 on page 534

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

Sample Output

request security idp ssl-inspection key add key1 file /var/tmp/enc1.key password encrypted

```
user@host> request security idp ssl-inspection key add key1 file /var/tmp/enc1.key password
encrypted
Added key 'key1'
```

Sample Output

request security idp ssl-inspection key add key2 file /var/tmp/enc2.key password encrypted

```
user@host> request security idp ssl-inspection key add key2 file /var/tmp/enc2.key password
encrypted
Added key 'key2', server 2.2.0.1
```

Sample Output

request security idp ssl-inspection key add key3 file /var/tmp/norm.key

```
user@host> request security idp ssl-inspection key add key3 file /var/tmp/norm.key
Added key 'key3'
```

Sample Output

request security idp ssl-inspection key add key1 server 1.1.0.1

```
user@host> request security idp ssl-inspection key add key1 server 1.1.0.1
Added key 'key1', server 1.1.0.1
```

Sample Output

request security idp ssl-inspection key add key1 server 1.1.0.2

```
user@host> request security idp ssl-inspection key add key1 server 1.1.0.2
Added key 'key1', server 1.1.0.2
```

request security idp ssl-inspection key delete

Supported Platforms SRX5400, SRX5600, SRX5800, vSRX

Syntax request security idp ssl-inspection key delete [<*key-name*> [server <*server-ip*>]]

Release Information Command introduced in Junos OS Release 9.3.

Description Delete the specified server IP from the given key if the server is specified. If the server IP is not specified, the given key will be deleted along with all the server addresses associated with it.

> NOTE: You will get a delete confirmation question before deleting one or more keys or server.

Options

- *key-name*—(Optional) Name of the SSL private key.
- server <*server-ip*> —(Optional) Server IP address associated with the specified key to be deleted.

Required Privilege Level maintenance

Related Documentation

- show security idp ssl-inspection key on page 591

List of Sample Output request security idp ssl-inspection key delete on page 535
request security idp ssl-inspection key delete key1 on page 536
request security idp ssl-inspection key delete key2 server 2.2.0.1 on page 536

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

Sample Output

request security idp ssl-inspection key delete

```
user@host> request security idp ssl-inspection key delete

This command will delete one or more ssl keys.
Continue? [yes,no] (no) yes

Number of keys 4, server 3 deleted
```

Sample Output

request security idp ssl-inspection key delete key1

```
user@host> request security idp ssl-inspection key delete key1

This command will delete one or more ssl keys.
Continue? [yes,no] (no) yes

Number of keys 1, server 2 deleted
```

Sample Output

request security idp ssl-inspection key delete key2 server 2.2.0.1

```
user@host> request security idp ssl-inspection key delete key2 server 2.2.0.1

This command will delete one or more ssl keys.
Continue? [yes,no] (no) yes

Number of keys 0, server 1 deleted
```

request security idp storage-cleanup

Supported Platforms SRX Series, vSRX

Syntax request security idp storage-cleanup

Release Information Command introduced in Junos OS Release 11.4.

Description Delete unused files to free up storage space on a device.

Options **cache-files**— Delete DFA cache files used for optimizing idp policy compilation.

downloaded-files— Delete downloaded security-package files (with out affecting the installed database).

Required Privilege Level maintenance

List of Sample Output request security idp storage-cleanup on page 537

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

Sample Output

request security idp storage-cleanup

```
user@host> request security idp storage-cleanup downloaded-files
Successfully deleted downloaded secdb files
```

show class-of-service forwarding-class

Supported Platforms SRX Series, vSRX

Syntax show class-of-service forwarding-class

Release Information Command introduced before Junos OS Release 12.1.

Description Display mapping of forwarding class names to queues.

Required Privilege Level view

Related Documentation

- Forwarding Classes Overview on page 235

List of Sample Output show class-of-service forwarding-class on page 538

Output Fields Table 97 on page 538 lists the output fields for the **show class-of-service forwarding-class** command. Output fields are listed in the approximate order in which they appear.

Table 97: show class-of-service forwarding-class Output Fields

Field Name	Field Description
Forwarding class	Forwarding class name.
ID	ID number assigned to the forwarding class.
Queue	Queue number.
Restricted queue	Restricted queue number.
Fabric priority	Fabric priority, either low or high.
Policing priority	Layer 2 policing, either premium or normal.
SPU priority	Services Processing Unit (SPU) priority queue, either high or low.

Sample Output

show class-of-service forwarding-class

```
user@host> show class-of-service forwarding-class
Forwarding class        ID  Queue  Restricted queue  Fabric priority  Policing
priority  SPU priority
  best-effort            0    0          0             low                  normal
          low
  expedited-forwarding   1    1          1             low                  normal
          high
```

```
assured-forwarding      2   2          2             low                 normal
        low
network-control         3   3          3             low                 normal
        low
```

show class-of-service rewrite-rule

Supported Platforms MX Series, NFX Series, PTX Series, SRX Series, vSRX

Syntax

```
show class-of-service rewrite-rule
<name name>
<type type>
```

Release Information Command introduced before Junos OS Release 7.4.

Description Display the mapping of forwarding classes and loss priority to code point values.

Options **none**—Display all rewrite rules.

name ***name***—(Optional) Display the specified rewrite rule.

type ***type***—(Optional) Display the rewrite rule of the specified type. The rewrite rule type can be one of the following:

- **dscp**—For IPv4 traffic.
- **dscp-ipv6**—For IPv6 traffic.
- **exp**—For MPLS traffic.
- **frame-relay-de**—(SRX Series only) For Frame Relay traffic.
- **ieee-802.1**—For Layer 2 traffic.
- **inet-precedence**—For IPv4 traffic.

Required Privilege Level view

Related Documentation

- Rewrite Rules Overview on page 237

List of Sample Output show class-of-service rewrite-rule type dscp on page 541

Output Fields Table 98 on page 540 describes the output fields for the **show class-of-service rewrite-rule** command. Output fields are listed in the approximate order in which they appear.

Table 98: show class-of-service rewrite-rule Output Fields

Field Name	Field Description
Rewrite rule	Name of the rewrite rule.
Code point type	Type of rewrite rule: **dscp**, **dscp-ipv6**, **exp**, **frame-relay-de**, or **inet-precedence**.

Table 98: show class-of-service rewrite-rule Output Fields (continued)

Field Name	Field Description
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router or switch.
Index	Internal index for this particular rewrite rule.
Loss priority	Loss priority for rewriting.
Code point	Code point value to rewrite.

Sample Output

show class-of-service rewrite-rule type dscp

```
user@host> show class-of-service rewrite-rule type dscp
Rewrite rule: dscp-default, Code point type: dscp
  Forwarding class                    Loss priority       Code point
  gold                                high                000000
  silver                              low                 110000
  silver                              high                111000
  bronze                              low                 001010
  bronze                              high                001100
  lead                                high                101110

Rewrite rule: abc-dscp-rewrite, Code point type: dscp, Index: 3245
Forwarding class                    Loss priority       Code point
  gold                                low                 000111
  gold                                high                001010
  silver                              low                 110000
  silver                              high                111000
  bronze                              high                001100
  lead                                low                 101110
  lead                                high                110111
```

show security flow session idp family

Supported Platforms SRX Series, vSRX

Syntax show security flow session idp family (inet | inet6)

Release Information Command introduced in Junos OS Release 10.2.
Support for family inet6 added in Junos OS Release 12.1X46-D10.

Description Display filtered summary of information about existing sessions, including types of sessions, active and failed sessions, and the maximum allowed number of sessions.

Options **inet**—Display details summary of IPv4 sessions.

inet6—Display details summary of IPv6 sessions.

Required Privilege Level view

Related Documentation

- Understanding Intrusion Detection and Prevention for SRX Series on page 27

List of Sample Output **show security flow session summary family inet on page 542**
show security flow session summary family inet6 on page 543

Output Fields Table 99 on page 542 lists the output fields for the **show security flow session summary family** command. Output fields are listed in the approximate order in which they appear.

Table 99: show security flow session summary Output Fields

Field Name	Field Description
Valid sessions	Count of valid sessions.
Pending sessions	Count of pending sessions.
Invalidated sessions	Count of sessions the security device has determined to be invalid.
Sessions in other states	Count of sessions not in valid, pending, or invalidated state.
Total sessions	Total of the above counts.

Sample Output

show security flow session summary family inet

```
user@host> show security flow session summary family inet
```

```
Flow Sessions on FPC4 PIC0:
Valid sessions: 3
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 3

Flow Sessions on FPC5 PIC0:
Valid sessions: 4
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 4
```

show security flow session summary family inet6

```
user@host> show security flow session summary family inet6

Flow Sessions on FPC1 PIC1:
Valid sessions: 20
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 20
```

show security flow session idp summary

Supported Platforms SRX Series, vSRX

Syntax **show security flow session idp summary**

Release Information Command introduced in Junos OS Release 10.2.

Description Display summary output.

Options

- application—Application name
- destination-port—Destination port
- destination-prefix—Destination IP prefix or address
- family—Display session by family.
- interface–Name of incoming or outgoing interface
- protocol—IP protocol number
- source-port—Source port
- source-prefix—Source IP prefix

Required Privilege Level view

Related Documentation

- *show security flow session*

List of Sample Output show security flow session idp summary on page 545

Output Fields Table 100 on page 544 lists the output fields for the **show security flow session idp summary** command. Output fields are listed in the approximate order in which they appear.

Table 100: show security flow session idp summary Output Fields

Field Name	Field Description
Valid session	Number of valid sessions.
Pending sessions	Number of pending sessions.
Invalidated sessions	Number of invalid sessions.
Sessions in other states	Number of sessions in other states.
Total sessions	Total number of sessions.

Sample Output

show security flow session idp summary

```
root@ show security flow session idp summary
Flow Sessions on FPC4 PIC0:

Valid sessions: 3
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 3

Flow Sessions on FPC5 PIC0:

Valid sessions: 4
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 4
```

show security idp active-policy

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp active-policy
```

Release Information Command introduced in Junos OS Release 9.2.

Description Display information about the policy name and running detector version with which the policy is compiled from the IDP data plane module.

Required Privilege Level view

Related Documentation

- request security idp security-package download on page 527
- request security idp security-package install on page 530

List of Sample Output show security idp active-policy on page 546

Output Fields Table 101 on page 546 lists the output fields for the **show security idp active-policy** command. Output fields are listed in the approximate order in which they appear.

Table 101: show security idp active-policy Output Fields

Field Name	Field Description
Policy Name	Name of the running policy.
Running Detector Version	Current version of the running detector.

Sample Output

show security idp active-policy

```
user@host> show security idp active-policy
  Policy Name : viking-policy
  Running Detector Version : 9.1.140080300
```

show security idp attack description

Supported Platforms SRX Series, vSRX

Syntax **show security idp attack description *attack-name***

Release Information Command introduced in Junos OS Release 11.4.

Description Display description of a specified IDP attack.

Options

- ***attack-name*** —IDP attack name.

Required Privilege Level view

Related Documentation

- clear security idp attack table on page 515

List of Sample Output show security idp attack description on page 547

Output Fields Table 102 on page 547 lists the output fields for the **show security idp attack description** command. Output fields are listed in the approximate order in which they appear.

Table 102: show security idp attack description Output Fields

Field Name	Field Description
Description	IDP attack description.

Sample Output

show security idp attack description

```
user@host> show security idp attack description FTP:USER:ROOT
Description: This signature detects attempts to login to an FTP server using the
 "root" account.  This can indicate an attacker trying to gain root-level access,
 or it can indicate poor security practices. FTP typically uses plain-text
passwords, and using the root account to FTP could expose sensitive data over the
 network.
```

show security idp attack detail

Supported Platforms SRX Series, vSRX

Syntax **show security idp attack detail *attack-name***

Release Information Command introduced in Junos OS Release 11.4.

Description Display details of a specified IDP attack.

Options

- ***attack-name*** —IDP attack name.

Required Privilege Level view

Related Documentation

- clear security idp attack table on page 515

List of Sample Output show security idp attack detail FTP:USER:ROOT on page 549
show security idp attack detail TROJAN:MISC:ROOTBEER-CLIENT on page 550

Output Fields Table 103 on page 548 lists the output fields for the **show security idp attack detail** command. Output fields are listed in the approximate order in which they appear.

Table 103: show security idp attack detail Output Fields

Field Name	Field Description
Display Name	Display name of the IDP attack.
Severity	Severity level of the IDP attack.
Category	IDP attack category.
Recommended	Specifies whether a default action for the IDP attack is recommended by Juniper Networks (true or false).
Recommended Action	Recommended action for the IDP attack.
Type	Type of IDP attack.
Direction	Direction of the IDP attack.
False Positives	Specifies whether the IDP attack produces false positive on the network.
Service	IDP service configured for the IDP attack. If a service is configured for the IDP attack, the IDP service name is displayed. Otherwise, **Not available** is displayed.

Table 103: show security idp attack detail Output Fields (continued)

Field Name	Field Description
Member Name	Name of attack member in IDP attack
Expression	Specifies the Boolean expression of attack members used to identify the way(for example, OR, AND, or oAND) attack members should be matched.
PCRE Expression	Specifies the Boolean expression of PCRE format based attack members used to identify the way(for example, OR, AND, or oAND) attack members should be matched. If this field is not present “Expression” is used as a Boolean expression for attack matching.
Shellcode	Signifies if the IDP attack is a shellcode attack.
Flow	Signifies the channel(control, data) of IDP attack.
Context	Name of the context under which IDP attack has to be matched.
Negate	Signifies if the signature in the IDP attack is a negate signature.
TimeBinding	Specifies count and scope under which the attack is valid.
Pattern	Specifies the regular expression in the IDP attack.
PCRE Pattern	Specifies the regular expression in PCRE format in the IDP attack.
Hidden Pattern	Specifies if the attack pattern is hidden.

Sample Output

show security idp attack detail FTP:USER:ROOT

```
user@hostt> run show security idp attack detail FTP:USER:ROOT
Display Name: FTP: "root" Account Login
Severity: Minor
Category: FTP
Recommended: false
Recommended Action: None
Type: signature
Direction: CTS
False Positives: unknown
Shellcode: no
Flow: control
Context: ftp-username
Negate: false
TimeBinding:
   Scope: none
        Count: 1
Hidden Pattern: False
Pattern: \[root\]
```

show security idp attack detail TROJAN:MISC:ROOTBEER-CLIENT

```
user@host> show security idp attack detail TROJAN:MISC:ROOTBEER-CLIENT
Display Name: TROJAN: Digital Rootbeer Client Connect
Severity: Minor
Category: TROJAN
Recommended: false
Recommended Action: None
Type: chain
False Positives: unknown
Service: TCP/2600
Expression: m01 oAND m02
Order: no
Reset: no
Scope: session
TimeBinding:
Members:
        Member Name: m01
        Type: Signature
        Direction: CTS
        Flow: control
        Shellcode: no
        Context: stream256
        Negate: false
        Hidden Pattern: False
        Pattern: .*/QUE,who are you\.\.\.\?.*
        PCRE Pattern: ^(.)*\/QUE,who are you\.\.\.\?

        Member Name: m02
        Type: Signature
        Direction: STC
        Flow: control
        Shellcode: no
        Context: stream256
        Negate: false
        Hidden Pattern: False
        Pattern: .*/QUE,billy the kid.*
        PCRE Pattern: ^(.)*\/QUE,billy the kid
```

show security idp attack table

Supported Platforms SRX Series, vSRX

Syntax show security idp attack table

Release Information Command introduced in Junos OS Release 9.2.

Description Display detailed information of IDP attack table.

Required Privilege Level view

Related Documentation

- clear security idp attack table on page 515

List of Sample Output show security idp attack table on page 551

Output Fields Table 104 on page 551 lists the output fields for the **show security idp attack table** command. Output fields are listed in the approximate order in which they appear.

Table 104: show security idp attack table Output Fields

Field Name	Field Description
Attack name	Name of the attack that you want to match in the monitored network traffic.
Hits	Total number of attack matches. On SRX Series devices, for brute force and time-binding-related attacks, the logging is to be done only when the match **count** is equal to the **threshold**. That is, only one log is generated within the 60-second period in which the threshold is measured. This process prevents repetitive logs from being generated and ensures consistency with other IDP platforms, such as IDP-standalone. When no attack is seen within the 60-second period and the BFQ entry is flushed out, the match count starts over the new attack match shows up in the attack table, and the log is generated.

Sample Output

show security idp attack table

```
user@host> show security idp attack table
IDP attack statistics:
  Attack name                                  #Hits
  HTTP:OVERFLOW:PI3WEB-SLASH-OF                1
```

show security idp counters application-identification

Supported Platforms SRX Series, vSRX

Syntax show security idp counters application-identification

Release Information Command introduced in Junos OS Release 9.2. Modified in Junos OS Release 12.1.

Description Display the status of all IDP application identification (AI) counter values.

Required Privilege Level view

Related Documentation

- clear security idp counters application-identification on page 516

List of Sample Output show security idp counters application-identification on page 554

Output Fields Table 105 on page 552 lists the output fields for the **show security idp counters application-identification** command. Output fields are listed in the approximate order in which they appear.

Table 105: show security idp counters application-identification Output Fields

Field Name	Field Description
AI matches	Number of sessions with an AI signature match.
AI no-matches	Number of sessions with no AI signature match.
AI-enabled sessions	Number of sessions with AI enabled.
AI-disabled sessions	Number of sessions with AI disabled.
AI-disabled sessions due to ssl encapsulated flows	Number of sessions with AI disabled due to SSL encapsulated flows.
AI-disabled sessions due to cache hit	Number of sessions with AI disabled due to a cache match.
AI-disabled sessions due to configuration	Number of sessions with AI disabled because the configured session limit was reached.
AI-disabled sessions due to protocol remapping	Number of sessions with AI disabled due to protocol remapping.
AI-disabled sessions due to RPC match	Number of sessions with AI disabled due to an RPC match.
AI-disabled sessions due to gate match	Number of sessions with AI disabled due to a gate match.

Table 105: show security idp counters application-identification Output Fields (continued)

Field Name	Field Description
AI-disabled sessions due to non-TCP/UDP flows	Number of sessions with AI disabled due to non-TCP or non-UDP flows.
AI-disabled sessions due to session limit	Number of sessions with AI disabled because the maximum session limit was reached.
AI-disabled sessions due to session packet memory limit	Number of sessions with AI disabled because the memory usage limit per session was reached.
AI-disabled sessions due to global packet memory limit	Number of sessions with AI disabled because the global memory usage limit was reached.
AI sessions current global reass packet memory usage	Number of AI sessions with current global reassembler packet memory usage limit
AI sessions peak global reass packet memory usage	Number of AI sessions with peak global reassembler packet memory usage limit
AI sessions current global packet memory usage	Number of AI sessions with current global packet memory usage limit
AI sessions peak global packet memory usage	Number of AI sessions with peak global packet memory usage limit
AI-sessions dropped due to malloc failure before session create	Number of AI sessions dropped because the malloc failure occurred before session create.
AI-sessions dropped due to malloc failure after create	Number of AI sessions dropped because the malloc failure occurred after session create.
AI-Packets received on sessions marked for drop due to malloc failure	Number of AI packets received on sessions that are marked to be dropped because the malloc failure.
Packets cloned for AI	Number of packets cloned for application identification.
Policy update	Number of times the IDP policy has been updated.
Total PME prematch job ignored	Number of jobs ignored because of pattern matching engine (PME) not matching.
Total packets for which prematch job were ignored	Number of packets for which signature matching was ignored as prematch found.
Prematch busy packet count	Number of packets saved as they are handed off for signature matching during prematch reprocess.
Final match busy packet count	Number of packets saved as they are handed off for signature matching during final match reprocess.
Total AI busy packet count	Number of times AI saved packet handed off for signature matching.

Table 105: show security idp counters application-identification Output Fields (continued)

Field Name	Field Description
Final match processed busy packet count	Number of times a packet processed for final matching before signature matching.
Prematch processed busy packet count	Number of times a packet processed for prematch before signature match.
Prematch ignored busy packet count	Number of packets ignored for signature matching as prematch found.
AI done busy packet count	Number of packets signature matching not completed before AI done.
JPME flow for Ignored jobs destroyed	Number of jobs destroyed because of flow mismatch due to policy relookup.
Set AI done for prematch	Number of sessions set for AI applied.
AI done for prematch	Number of sessions with AI applied.

Sample Output

show security idp counters application-identification

```
user@host> show security idp counters application-identification

IDP counter type                                                      Value
 AI matches                                                              0
 AI no-matches                                                           0
 AI-enabled sessions                                                     0
 AI-disabled sessions                                                    0
 AI-disabled sessions due to ssl encapsulated flows                      0
 AI-disabled sessions due to cache hit                                   0
 AI-disabled sessions due to configuration                               0
 AI-disabled sessions due to protocol remapping                          0
 AI-disabled sessions due to RPC match                                   0
 AI-disabled sessions due to gate match                                  0
 AI-disabled sessions due to non-TCP/UDP flows                           0
 AI-disabled sessions due to session limit                               0
 AI-disabled sessions due to session packet memory limit                 0
 AI-disabled sessions due to global packet memory limit                  0
 AI sessions current global reass packet memory usage                    0
 AI sessions peak global reass packet memory usage                       0
 AI sessions current global packet memory usage                          0
 AI sessions peak global packet memory usage                             0
 AI-sessions dropped due to malloc failure before session create         0
 AI-sessions dropped due to malloc failure after create                  0
 AI-Packets received on sessions marked for drop due to malloc failure   0
 Packets cloned for AI                                                   0
 Policy update                                                           0
 Total PME prematch job ignored                                          0
 Total packets for which prematch job were ignored                       0
 Prematch busy packet count                                              0
 Final match busy packet count                                           0
 Total AI busy packet count                                              0
 Final match processed busy packet count                                 0
```

```
Prematch processed busy packet count                                    0
Prematch ignored busy packet count                                      0
AI done busy packet count                                               0
JPME flow for Ignored jobs destroyed                                    0
Set AI done for prematch                                                0
AI done for prematch                                                    0
                                                                        0
```

show security idp counters dfa

Supported Platforms SRX Series, vSRX

Syntax show security idp counters dfa

Release Information Command introduced in Junos OS Release 9.2.

Description Display the status of all DFA counter values.

Required Privilege Level view

Related Documentation

- clear security idp counters dfa on page 517

List of Sample Output show security idp counters dfa on page 556

Output Fields Table 106 on page 556 lists the output fields for the **show security idp counters dfa** command. Output fields are listed in the approximate order in which they appear.

Table 106: show security idp counters dfa Output Fields

Field Name	Field Description
DFA Group Merged Usage	Number of DFA groups merged.
DFA Matches	Number of DFA matches found.

Sample Output

show security idp counters dfa

```
user@host> show security idp counters dfa
IDP counters:
IDP counter type                                                     Value
 DFA Group Merged Usage                                                0
 DFA Matches                                                           1
```

show security idp counters flow

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp counters flow
```

Release Information Command introduced in Junos OS Release 9.2.

Description Display the status of all IDP flow counter values.

> NOTE: On SRX Series devices with IDP enabled, if IDP attacks are configured for a single direction (server or client), a flow in the opposite direction does not need IDP processing. For TCP traffic, the TCP optimization feature ensures minimal processing for these flows without running into reassembly errors.

Required Privilege Level view

Related Documentation

- flow (Security IDP) on page 378
- clear security idp counters flow on page 518

List of Sample Output show security idp counters flow on page 561

Output Fields Table 107 on page 557 lists the output fields for the **show security idp counters flow** command. Output fields are listed in the approximate order in which they appear.

Table 107: show security idp counters flow Output Fields

Field Name	Description
Fast-path packets	Number of packets that are set through fast path after completing IDP policy lookup.
Slow-path packets	Number of packets that are sent through slow path during IDP policy lookup.
Session construction failed (Unsupported)	Number of times the packet failed to establish the session.
Session limit reached	Number of sessions that reached IDP sessions limit.
Session inspection depth reached	Number of sessions that reached inspection depth.
Memory limit reached	Number of sessions that reached memory limit.

Table 107: show security idp counters flow Output Fields (continued)

Field Name	Description
Not a new session (Unsupported)	Number of sessions that extended beyond time limit.
Invalid index at age-out (Unsupported)	Invalid session index in session age-out message.
Packet logging	Number of packets saved for packet logging.
Policy cache hits	Number of sessions that matched policy cache.
Policy cache misses	Number of sessions that did not match policy cache.
Policy cache entries	Number of policy cache entries.
Maximum flow hash collisions	Maximum number of packets, of one flow, that share the same hash value.
Flow hash collisions	Number of packets that share the same hash value.
Gates added	Number of gate entries added for dynamic port identification.
Gate matches (Unsupported)	Number of times a gate is matched.
Sessions deleted	Number of sessions deleted.
Sessions aged-out (Unsupported)	Number of sessions that are aged out if no traffic is received within session timeout value.
Sessions in-use while aged-out (Unsupported)	Number of sessions in use during session age-out.
TCP flows marked dead on RST/FIN	Number of sessions marked dead on TCP RST/FIN.
policy init failed	Policy initiation failed.
Number of times Sessions exceed high mark	Number of times sessions exceeded the high mark.
Number of sessions exceeds high mark	Number of sessions that exceed high mark.
Number of sessions drops below low mark	Number of sessions that fall below low mark.

Table 107: show security idp counters flow Output Fields (continued)

Field Name	Description
Memory of sessions exceeds high mark	Session memory exceeds high mark.
Memory of sessions drops below low mark	Session memory drops below low mark.
SM Sessions encountered memory failures	Number of SM sessions that encountered memory failures.
SM Packets on sessions with memory failures	Number of SM packets that encountered memory failures.
Sessions constructed	Number of sessions established.
SM Sessions dropped	Number of SM sessions dropped.
SM sessions ignored	Number of sessions ignored in Security Module (SM).
SM sessions interested	Number of SM sessions interested.
SM sessions not interested	Number of SM sessions not interested.
SM sessions interest error	Number of errors created for SM sessions interested.
Sessions destructed	Number of sessions destructed.
SM Session Create	Number of SM sessions created.
SM Packet Process	Number of packets processed from SM.
SM FTP data session ignored by IDP	Number of SM FTP data sessions that are ignored by IDP.
SM Session close	Number of SM sessions closed.
SM client-to-server packets	Number of SM client-to-server packets.
SM server-to-client packets	Number of SM server-to-client packets.
SM client-to-server L7 bytes	Number of SM client-to-server Layer 7 bytes.
SM server-to-client L7 bytes	Number of SM server-to-client Layer 7 bytes.
Client-to-server flows ignored	Number of client-to-server flow sessions that are ignored.
Server-to-client flows ignored	Number of server-to-client flow sessions that are ignored.
Server-to-client flows tcp optimized	Number of server-to-client flow TCP sessions that are optimized.

Table 107: show security idp counters flow Output Fields (continued)

Field Name	Description
Client-to-server flows tcp optimized	Number of client-to-server flow TCP sessions that are optimized.
Both directions flows ignored	Number of server-to-client and client-to-server flow sessions that are ignored.
Fail-over sessions dropped	Number of failover sessions dropped.
Sessions dropped due to no policy	Number of sessions dropped because there was no active IDP policy.
IDP Stream Sessions dropped due to memory failure	Number of IDP stream sessions that are dropped because of memory failure.
IDP Stream Sessions ignored due to memory failure	Number of IDP stream sessions that are ignored because of memory failure.
IDP Stream Sessions closed due to memory failure	Number of IDP stream sessions that are closed because of memory failure.
IDP Stream Sessions accepted	Number of IDP stream sessions that are accepted.
IDP Stream Sessions constructed	Number of IDP stream sessions that are constructed.
IDP Stream Sessions destructed	Number of IDP stream sessions that are destructed.
IDP Stream Move Data	Number of stream data events handled by IDP.
IDP Stream Sessions ignored on JSF SSL Event	Number of IDP stream sessions that are ignored because of a JSF SSL proxy event.
IDP Stream Sessions not processed for no matching rules	Number of IDP stream sessions that are not processed for no matching rules.
IDP Stream stbuf dropped	Number of IDP stream plug-in buffers dropped.
IDP Stream stbuf reinjected	Number of IDP stream plug-in buffers injected.
Busy packets from stream plugin	Number of packets saved as one or more packets of this session from stream plug-in.
Busy packets from packets plugin	Number of saved packets for IDP stream plug-in sessions.
Bad kpp	Number of internal marked packets logged for IDP processing.
Lsys policy id lookup failed sessions	Number of sessions that failed logical systems policy lookup.
Busy packets	Number of packets saved as one or more packets of this session are handed off for asynchronous processing.
Busy packet errors	Number of packets found with IP checksum error after asynchronous processing is completed.

Table 107: show security idp counters flow Output Fields (continued)

Field Name	Description
Dropped queued packets (async mode)	Number of queued packets dropped based on policy action, reinjection failures, or if the session is marked to destruct.
Dropped queued packets failed (async mode)	Not used currently.
Reinjected packets (async mode)	Number of packets reinjected into the queue.
Reinjected packets failed(async mode)	Number of failed reinjected packets.
AI saved processed packet	Number of AI packets saved for which the asynchronous processing is completed.
Busy packet count incremented	Number of times the busy packet count incremented in asynchronous processing.
busy packet count decremented	Number of times the busy packet count decremented in asynchronous processing.
session destructed in pme	Number of sessions destructed as a part of asynchronous result processing.
session destruct set in pme	Number of sessions set to be destructed as a result of asynchronous processing.
KQ op	Number of sessions with one of the following status: • KQ op hold—number of times packets held by IDP. • KQ op drop—number of times packets dropped by IDP. • KQ op route—number of times IDP decided to be route the packet directly. • KQ op Continue—number of times IDP decided to continue to process the packet. • KQ op error—number of times error occurred while IPD processing packet. • KQ op stop—number of times IDP decided to stop processing the packet.
PME wait not set	Number of AI saved packets given for signature matching.
PME wait set	Number of packets given for signature matching without AI save.
PME KQ run not called	Number of times signature matching results processed out of packet receiving order.

Sample Output

show security idp counters flow

```
user@host> show security idp counters flow
IDP counters:

  IDP counter type                                                         Value
 Fast-path packets                                                          40252
 Slow-path packets                                                          127
 Session construction failed                                                0
 Session limit reached                                                   0
 Session inspection depth reached                                        0
 Memory limit reached                                                    0
 Not a new session                                                       0
 Invalid index at ageout                                                 0
 Packet logging                                                          0
 Policy cache hits                                                       92
 Policy cache misses                                                     67
 Policy cache entries                                                    67
 Maximum flow hash collisions                                            0
 Flow hash collisions                                                    0
 Gates added                                                             0
 Gate matches                                                            0
 Sessions deleted                                                        127
 Sessions aged-out                                                       0
 Sessions in-use while aged-out                                          0
 TCP flows marked dead on RST/FIN                                        13
 Policy init failed                                                      0
 Number of times Sessions exceed high mark                               0
 Number of times Sessions drop below low mark                            0
 Memory of Sessions exceeds high mark                                    0
 Memory of Sessions drops below low mark                                 0
 SM Sessions encountered memory failures                                 0
 SM Packets on sessions with  memory failures                            0
 IDP session gate creation requests                                      0
 IDP session gate creation acknowledgements                              0
 IDP session gate hits                                                   0
 IDP session gate timeouts                                               0
 Number of times Sessions crossed the CPU threshold value that is set    0
 Number of times Sessions crossed the CPU upper threshold                0
 Sessions constructed                                                    127
 SM Sessions ignored                                                     0
 SM Sessions dropped                                                     0
 SM Sessions interested                                                  168
 SM Sessions not interested                                              4
 SM Sessions interest error                                              0
 Sessions destructed                                                     127
 SM Session Create                                                       127
 SM Packet Process                                                       52257
 SM ftp data session ignored by idp                                      0
 SM Session close                                                        127
 SM Client-to-server packets                                             20066
 SM Server-to-client packets                                             32191
 SM Client-to-server L7 bytes                                            167292
 SM Server-to-client L7 bytes                                           28523514

 Client-to-server flows ignored                                          1
 Server-to-client flows ignored                                          1
 Server-to-client flows tcp optimized                                    3
 Client-to-server flows tcp optimized                                    0
 Both directions flows ignored                                           32
 Fail-over sessions dropped                                              0
 Sessions dropped due to no policy                                       0
 IDP Stream Sessions dropped due to memory failure                       0
 IDP Stream Sessions ignored due to memory failure                       0
 IDP Stream Sessions closed due to memory failure                        0
 IDP Stream Sessions accepted                                            0
 IDP Stream Sessions constructed                                         0
 IDP Stream Sessions destructed                                          0
 IDP Stream Move Data                                                    0
 IDP Stream Sessions ignored on JSF SSL Event                            0
 IDP Stream Sessions not processed for no matching rules              0
 IDP Stream stbuf dropped                                             0
 IDP Stream stbuf reinjected                                          0
 Busy pkts from stream plugin                                         0
 Busy pkts from pkt plugin                                            0
 bad kpp                                                              0
 Lsys policy id lookup failed sessions                                0
 Busy packets                                                         0
 Busy packet Errors                                                   0
 Dropped queued packets (async mode)                                  0
 Dropped queued packets failed(async mode)                            0
 Reinjected packets (async mode)                                      0
 Reinjected packets failed(async mode)                                0
 AI saved processed packet                                            0
 busy packet count incremented                                        0
 busy packet count decremented                                        0
 session destructed in pme                                            0
 session destruct set in pme                                          0
 kq op hold                                                           0
 kq op drop                                                           0
 kq op route                                                          0
 kq op continue                                                       35155
 kq op error                                                          0
 kq op stop                                                           0
 PME wait not set                                                     0
 PME wait set                                                         0
 PME KQ run not called                                                0
```

show security idp counters http-decoder

Supported Platforms SRX Series, vSRX

Syntax show security idp counters http-decoder

Release Information Command introduced in Junos OS Release 11.2.

Description Display the status of all HTTP decoders.

Required Privilege Level view

Related Documentation

- clear security idp counters http-decoder on page 519

List of Sample Output show security idp counters http-decoder on page 564

Output Fields Table 108 on page 564 lists the output fields for the **show security idp counters http-decoder** command. Output fields are listed in the approximate order in which they appear.

Table 108: show security idp counters http-decoder Output Fields

Field Name	Field Description
No of file-decoder requests from MIME over HTTP	Number of active file decoder requests sent over HTTP from MIME.
No of pending file-decoder requests from MIME over HTTP	Number of pending file decoder requests sent over HTTP from MIME.
No of completed file-decoder requests from MIME over HTTP	Number of completed file decoder requests sent over HTTP from MIME.
No of unrecognized file type from MIME over HTTP	Number of unrecognized file types sent over HTTP from MIME.
No of compressed payload transferred over HTTP	Number of compressed files transferred over HTTP from MIME.

Sample Output

show security idp counters http-decoder

```
user@host> show security idp counters http-decoder
IDP counters:
IDP counter type                                                   Value
No of file-decoder requests from MIME over HTTP                        0
No of pending file-decoder requests from MIME over HTTP                0
No of completed file-decoder requests from MIME over HTTP              0
No of unrecognized file type from MIME over HTTP                         0
No of compressed payload transferred over HTTP                           0
```

show security idp counters ips

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp counters ips
```

Release Information Command modified in Junos OS Release 11.2.

Description Display the status of all IPS counter values.

Required Privilege Level view

Related Documentation

- ips on page 407
- clear security idp counters ips on page 520

List of Sample Output show security idp counters ips on page 567

Output Fields Table 109 on page 566 lists the output fields for the **show security idp counters ips** command. Output fields are listed in the approximate order in which they appear.

Table 109: show security idp counters ips Output Fields

Field Name	Field Description
TCP fast path	Number of TCP packets skipped for IDS processing.
Layer-4 anomalies	Number of Layer-4 protocol error or anomaly.
Anomaly hash misses	Number of times look failed on anomaly hash.
Line context matches	Number of attempts to match line based attacks in traffic stream.
Stream256 context matches	Number of attempts to match stream based attacks in first 256 bytes of traffic stream.
Stream context matches	Number of attempts to match stream based attacks in traffic stream.
Packet context matches	Number of attempts to match packet based attacks in traffic packet.
Packet header matches	Number of attempts to match packet header based attacks in traffic packet.
Context matches	Number of attempts to match protocol context based attacks in traffic stream.
Regular expression matches	Number of attempts to match PCRE expressions in traffic stream.
Tail DFAs	Number of attempts to match an attack on tail DFA group matches.

Table 109: show security idp counters ips Output Fields (continued)

Field Name	Field Description
Exempted attacks	Number of attacks exempted from match as per exempt rulebase.
Out of order chains	Number of times attack is excluded from match due to member attacks in an attack group did not complete chain.
Partial chain matches	Number of attacks in partial chain match with attack scope as transaction.
IDS device FIFO size	Number of IDS contexts in virtual IDS device.
IDS device FIFO overflows	Number of times an IDS context can not be written as the IDS device is full.
Brute force queue size	Number of entries in the brute force queue.
IDS cache hits (Unsupported)	Number of sessions those found attack instance in IDS cache.
IDS cache misses (Unsupported)	Number of sessions those did not find attack instance in IDS cache.
Shellcode detection invocations	Number of times shell code match is attempted.
Wrong offsets	Number of times attack's offset is not within the service offset range.
No peer MAC (Unsupported)	Number of times flow peer MAC address is not available.

Sample Output

show security idp counters ips

```
user@host> show security idp counters ips
IDP counters:
 IDP counter type                                                      Value
 TCP fast path                                                           15
 Layer-4 anomalies                                                       0
 Anomaly hash misses                                                     3
 Line context matches                                                    5
 Stream256 context matches                                               5
 Stream context matches                                                  5
 Packet context matches                                                  0
 Packet header matches                                                   0
 Context matches                                                         12
 Regular expression matches                                              0
 Tail DFAs                                                               0
 Exempted attacks                                                        0
 Out of order chains                                                     0
 Partial chain matches                                                   0
 IDS device FIFO size                                                    0
 IDS device FIFO overflows                                               0
 Brute force queue size                                                  0
 IDS cache hits                                                          0
 IDS cache misses                                                        0
 Shellcode detection invocations                                         0
 Wrong offsets                                                           0
 No peer MAC                                                             0
 Content-decompression memory usage in KB                                0
 Content-decompression memory over limit                                 0
 Content-decompression gunzip called                                     0
 Content-decompression gunzip failed                                     0
 Content-decompression others called                                     0
 Content-decompression others failed                                     0
 Content-decompression input bytes                                       0
 Content-decompression output bytes                                      0
 Content-decompression ratio over limit                                  0
 Content-decompression type mismatch                                     0
```

show security idp counters log

Supported Platforms SRX Series, vSRX

Syntax show security idp counters log

Release Information Command introduced in Junos OS Release 9.2.

Description Display the status of all IDP log counter values.

Required Privilege Level view

Related Documentation

- *event-rate*
- *clear security idp counters log*

List of Sample Output show security idp counters log on page 571

Output Fields Table 110 on page 569 lists the output fields for the **show security idp counters log** command. Output fields are listed in the approximate order in which they appear.

Table 110: show security idp counters log Output Fields

Field Name	Field Description
Logs dropped	Number of logs that are dropped.
Suppressed log count	Number of logs that are suppressed.
Logs waiting for post-window packets (Unsupported)	Number of logs waiting for post-window packets.
Logs ready to be sent (Unsupported)	Number of logs ready to be sent.
Logs in suppression list (Unsupported)	Number of logs considered for suppression list.
Log timers created	Number of times the log timer is created.
Logs timers expired	Number of times the log timer is expired.
Log timers cancelled	Number of times the log timer is canceled.

Table 110: show security idp counters log Output Fields (continued)

Field Name	Field Description
Logs ready to be sent high watermark (Unsupported)	Number of packets that are ready to be sent with high degree watermark.
Log receive buffer full (Unsupported)	Number of times the buffer is full.
Packet log too big (Unsupported)	Number of packet logs that exceeded allowed packet log size.
Reads per second (Unsupported)	Number of packets that are read per second.
Logs in read buffer high watermark (Unsupported)	Number of high watermark packets that are in read buffer.
Packets logged	Number of packets that are logged,
Packets lost (Unsupported)	Number of packets that are failed to log.
Packets copied (Unsupported)	Number of packets copied during packet log.
Packets held (Unsupported)	Number of packets held for packet log.
Packets released	Number of packets that are released from hold.
IP Action Messages (Unsupported)	Number of IP action messages.
IP Action Drops (Unsupported)	Number of IP action messages dropped.
IP Action Exists (Unsupported)	Number of exits during IP action creation.
NWaits (Unsupported)	Number of logs waiting for post window packets.

Table 110: show security idp counters log Output Fields (continued)

Field Name	Field Description
Match vectors	Number of attacks in IDS match vector.
Supercedes	Number of attacks in supercede vector.

Sample Output

show security idp counters log

```
user@host> show security idp counters log
IDP counters:
 IDP counter type                                                      Value
 Logs dropped                                                            0
 Suppressed log count                                                    0
 Logs waiting for post-window packets                                    0
 Logs ready to be sent                                                   0
 Logs in suppression list                                                0
 Log timers created                                                      0
 Logs timers expired                                                     0
 Log timers cancelled                                                    0
 Logs ready to be sent high watermark                                    0
 Log receive buffer full                                                 0
 Packet log too big                                                      0
 Reads per second                                                        1
 Logs in read buffer high watermark                                      0
 Log Bytes in read buffer high watermark                                 0
 Packets logged                                                          0
 Packets lost                                                            0
 Packets copied                                                          0
 Packets held                                                            0
 Packets released                                                        0
 IP Action Messages                                                      0
 IP Action Drops                                                         0
 IP Action Exists                                                        0
 NWaits                                                                  0
 Match vectors                                                           0
 Supercedes                                                              0
 Kpacket too big                                                         0
```

show security idp counters packet

Supported Platforms SRX Series, vSRX

Syntax show security idp counters packet

Release Information Command introduced in Junos OS Release 9.2. The fields **Dropped by IDP policy** and **Dropped by Error** added in Junos OS Release 10.1.

Description Display the status of all IDP packet counter values.

Required Privilege Level view

Related Documentation

- clear security idp counters packet on page 522

List of Sample Output show security idp counters packet on page 574

Output Fields Table 111 on page 572 lists the output fields for the **show security idp counters packet** command. Output fields are listed in the approximate order in which they appear.

Table 111: show security idp counters packet Output Fields

Field Name	Field Description
Processed packets	Number of packets processed by the IDP service.
Dropped packets	Number of packets dropped by the IDP service. The counter for all dropped packets.
Dropped by IDP policy	Number of packets dropped by the IDP policy. The counter for dropped packets due to the action specified in the IDP policy (starting with the attack detection).
Dropped by Error	Number of packets dropped by error. The difference between **Dropped packets** and **Dropped by IDP policy**. IDS drops are primarily due to policy actions. Reassembly errors lead to packet drops. So all drops shown in **show security idp counters ips**, **show security idp counters flow** and **show security idp counters tcp-reassembler** add to **Dropped by Error**. All drops includes reassembly errors, anomalies similar to bad ip header and TTL errors.
Dropped sessions (Unsupported)	Number of sessions dropped.

Table 111: show security idp counters packet Output Fields (continued)

Field Name	Field Description
Bad IP headers	Number of packets that fail IP header length validity check.
Packets with IP options	Number of packets that contain the optional header fields.
Decapsulated packets	Number of packets that are decapsulated.
GRE decapsulations (Unsupported)	Number of packets that are generic routing encapsulation (GRE) decapsulated.
PPP decapsulations (Unsupported)	Number of packets that are Point-to-Point Protocol (PPP) decapsulated.
TCP decompression uncompressed IP (Unsupported)	Number of uncompressed IP headers that are to be TCP decompressed.
TCP decompression compressed IP (Unsupported)	Number of compressed IP headers that are to be TCP decompressed.
Deferred-send packets (Unsupported)	Number of deferred IP packets that are sent out.
IP-in-IP packets (Unsupported)	Number of packets that are IP-in-IP encapsulated.
TTL errors (Unsupported)	Number of packets with TTL error in the header.
Routing loops (Unsupported)	Number of packets that continue to be routed in an endless circle due to an inconsistent routing state.
No-route packets (Unsupported)	Number of packets that could not be routed further.
Flood IP (Unsupported)	Number of packets that are identified as IP flood packets.
Invalid ethernet headers (Unsupported)	Number of packets that are identified with an invalid Ethernet header.
Packets attached	Number of packets attached.

Table 111: show security idp counters packet Output Fields (continued)

Field Name	Field Description
Packets cloned	Number of packets that are cloned.
Packets allocated	Number of packets allocated.
Packets destructed	Number of packets destructed.

Sample Output

show security idp counters packet

```
user@host> show security idp counters packet
IDP counters:
 IDP counter type                                                      Value
 Processed packets                                                       27
 Dropped packets                                                         0
 Dropped by IDP policy                                                   0
 Dropped by error                                                        0
 Dropped sessions                                                        0
 Bad IP headers                                                          0
 Packets with IP options                                                 0
 Decapsulated packets                                                    0
 GRE decapsulations                                                      0
 PPP decapsulations                                                      0
 TCP decompression uncompressed IP                                       0
 TCP decompression compressed IP                                         0
 Deferred-send packets                                                   0
 IP-in-IP packets                                                        0
 TTL errors                                                              0
 Routing loops                                                           0
 STP drops                                                               0
 No-route packets                                                        0
 Flood IP                                                                0
 Invalid ethernet headers                                                0
 Packets attached                                                        28
 Packets cloned                                                          28
 Packets allocated                                                       0
 Packets destructed                                                      55
```

show security idp counters packet-log

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp counters packet-log
```

Release Information Command introduced in Junos OS Release 10.2.

Description Display the values of all IDP packet-log counters.

Required Privilege Level view

Output Fields The following table lists the output fields for the **show security idp counters packet-log** command. Output fields are listed in the approximate order in which they appear.

Field Name	Field Description
Total packets captured since packet capture was activated	Number of packets captured by the device by the IDP service.
Total sessions enabled since packet capture was activated	Number of sessions that have performed packet capture since the capture facility was activated.
Sessions currently enabled for packet capture	Number of sessions that are actively capturing packets at this time.
Packets currently captured for enabled sessions	Number of packets that have been captured by active sessions.
Packet clone failures	Number of packet capture failures due to cloning error.
Session log object failures	Number of objects containing log messages generated during packet capture that were not successfully transmitted to the host.
Session packet log object failures	Number of objects containing captured packets that were not successfully transmitted to the host.
Sessions skipped because session limit exceeded	Number of sessions that could not initiate packet capture because the maximum number of sessions specified for the device were conducting captures at that time.
Packets skipped because packet limit exceeded	Number of packets not captured because the packet limit specified for this device was reached.
Packets skipped because total memory limit exceeded	Number of packets not captured because the memory allocated for packet capture on this device was exceeded.

Sample Output

show security idp counters packet-log

```
user@host> show security idp counters packet-log
IDP counters:                                                       Value
 Total packets captured since packet capture was activated               0
 Total sessions enabled since packet capture was activated               0
 Sessions currently enabled for packet capture                           0
 Packets currently captured for enabled sessions                         0
 Packet clone failures                                                   0
 Session log object failures                                             0
 Session packet log object failures                                      0
 Sessions skipped because session limit exceeded                         0
 Packets skipped because packet limit exceeded                           0
 Packets skipped because total memory limit exceeded                     0
```

show security idp counters policy-manager

Supported Platforms SRX Series, vSRX

Syntax show security idp counters policy-manager

Release Information Command introduced in Junos OS Release 9.2.

Description Display the status of all IDP policies counter values.

Required Privilege Level view

Related Documentation

- clear security idp counters policy-manager on page 523

List of Sample Output show security idp counters policy-manager on page 577

Output Fields Table 112 on page 577 lists the output fields for the **show security idp counters policy-manager** command. Output fields are listed in the approximate order in which they appear.

Table 112: show security idp counters policy-manager Output Fields

Field Name	Field Description
Number of policies	Number of policies installed.
Number of aged out policies	Number of IDP policies that are expired.

Sample Output

show security idp counters policy-manager

```
user@host> show security idp counters policy-manager
IDP counters:
 IDP counter type                                                      Value
 Number of policies                                                      0
 Number of aged out policies                                             0
```

show security idp counters tcp-reassembler

Supported Platforms SRX Series, vSRX

Syntax show security idp counters tcp-reassembler

Release Information Command introduced in Junos OS Release 9.2.

Description Display the status of all TCP reassembler counter values.

NOTE: On SRX Series devices with IDP enabled, if IDP attacks are configured for a single direction (server or client), a flow in the opposite direction does not need IDP processing. For TCP traffic, the TCP optimization feature ensures minimal processing for these flows without running into reassembly errors.

Required Privilege Level view

Related Documentation

- re-assembler on page 450
- clear security idp counters tcp-reassembler on page 524

List of Sample Output show security idp counters tcp-reassembler on page 580

Output Fields Table 113 on page 578 lists the output fields for the **show security idp counters tcp-reassembler** command. Output fields are listed in the approximate order in which they appear.

Table 113: show security idp counters tcp-reassembler Output Fields

Field Name	Field Description
Bad TCP checksums (Unsupported)	Number of packets that have incorrect TCP checksums.
Bad TCP headers	Number of bad TCP headers detected.
Slow path segments	Number of segments that are sent through the slow path if the TCP segment does not pass fast-path segment validation.
Fast path segments	Number of segments that are sent through the fast path after passing a predefined TCP validation sequence.
Tcp Optimized s2c segments	Number of TCP segments that are sent through optimized re-assembly process from server to client.

Table 113: show security idp counters tcp-reassembler Output Fields (continued)

Field Name	Field Description
Tcp Optimized c2s segments	Number of TCP segments that are sent through optimized re-assembly process from server to client.
Sequence number wrap around errors	Number of packets that wrap around of the sequence number.
Session reuses	Number of sessions that reused an already established TCP session.
SYN retransmissions	Number of SYN packets that are retransmitted.
Bad three way handshake acknowledgements	Number of packets that have incorrect three-way handshake acknowledgements (ACK packet).
Sequence number out of sync flows	Number of packets that have out-of-sync sequence numbers.
Fast path pattern matches in queued up streams	Number of queued packets that have fast path pattern match.
New segments with no overlaps with old segment	Number of new segments that do not overlap with old segment.
New segment overlaps with beginning of old segment	Number of new segments that overlap with beginning of old segment.
New segment overlaps completely with old segment	Number of new segments that overlap completely with old segment.
New segment is contained in old segment	Number of new segments contained in old segment.
New segment overlaps with end of old segment	Number of new segments that overlap with the end of old segment.
New segment begins after end of old segment	Number of new segments that overlap after the end of old segment.
Memory consumed by new segment	Memory that is consumed by the new segment.
Peak memory consumed by new segments	Peak memory that is consumed by the new segment.
Segments in memory	Number of segments that are stored in memory for processing.
Per-flow memory overflows	Number of segments dropped after reaching per flow memory limit.
Global memory overflows	Number of segments dropped after reaching reassembler global memory limit.
Overflow drops	Number of packets that are dropped due to memory overflow.

Table 113: show security idp counters tcp-reassembler Output Fields (continued)

Field Name	Field Description
Copied packets (Unsupported)	Number of packets copied in reassembler.
Closed Acks	Number of Ack packets seen without having seen SYN on the same session.
Ack Validation failures	Number of Invalid ACKs received from server during 3-way handshake.
Simultanious syn	Number of simultaneous syn packets seen.
C2S synack	Number of C2S Syn/Ack packets seen.
Segment to left of receiver window	Number of segments falling left of receive window.
Segment to right of receiver window	Number of segments falling right of receive window.
SYN seen in the window	Number of Syn packets seen after connection establishment.
ACK bit is off	Number of packets seen without ACK after connection establishment.
Unexpected FIN	Number of unexpected FIN packets seen.
Duplicate Syn/Ack with different SEQ	Number of Syn/Ack packets with different SEQ numbers.

Sample Output

show security idp counters tcp-reassembler

```
user@host> show security idp counters tcp-reassembler
 IDP counters:

  IDP counter type                                                      Value
 Bad TCP checksums                                                       0
 Bad TCP headers                                                         0
 Slow path segments                                                      90
 Fast path segments                                                      7099
 Tcp Optimized s2c segments                                              0
 Tcp Optimized c2s segments                                              0
 Sequence number wrap around errors                                      0
 Session reuses                                                          0
 SYN retransmissions                                                     0
 Bad three way handshake acknowledgements                                0
 Sequence number out of sync flows                                       0
 Fast path pattern matches in queued up streams                          0
 New segments with no overlaps with old segment                          0
 New segment overlaps with beginning of old segment                      0
 New segment overlaps completely with old segment                        0
 New segment is contained in old segment                                 0
 New segment overlaps with end of old segment                            0
 New segment begins after end of old segment                             3
 Memory consumed by new segment                                          0
```

```
Peak memory consumed by new segments                                    3821
Segments in memory                                                      0
Per-flow memory overflows                                               0
Global memory overflows                                                 0
Overflow drops                                                          0
Copied packets                                                          0
Closed Acks                                                             3
Ack Validation failure                                                  0
Simultanious syn                                                        0
C2S synack                                                              0
segment to left of receiver window                                      0
segment to right of receiver window                                     0
SYN seen in the window                                                  0
ACK bit is off                                                          0
Unexpected FIN                                                          0
Duplicate Syn/Ack with different SEQ                                    0
```

show security idp logical-system policy-association

Supported Platforms SRX Series

Syntax show security idp logical-system policy-association

Release Information Command introduced in Junos OS Release 11.3.

Description Display the IDP policy assigned to a logical system. The IDP policy is assigned to a logical system through the security profile.

Required Privilege Level view

Related Documentation

- *security-profile*

List of Sample Output show security idp logical-system policy-association on page 582

Output Fields Table 114 on page 582 lists the output fields for the **show security idp logical-system policy-association** command.

Table 114: show security idp logical-system policy-association Output Fields

Field Name	Field Description
Logical system	Name of the logical system to which an IDP policy is assigned.
IDP policy	Name of the IDP policy that is specified in the security profile that is bound to the logical system.

Sample Output

show security idp logical-system policy-association

```
user@host> show security idp logical-system policy-association
Logical system         IDP policy
root-logical-system    idp-policy1
lsys1                  idp-policy2
```

show security idp memory

Supported Platforms SRX Series, vSRX

Syntax show security idp memory

Release Information Command introduced in Junos OS Release 9.2. Percentage outputs added in Junos OS Release 10.1.

Description Display the status of all IDP data plane memory.

Required Privilege Level view

List of Sample Output show security idp memory on page 583

Output Fields Table 115 on page 583 lists the output fields for the **show security idp memory** command. Output fields are listed in the approximate order in which they appear.

Table 115: show security idp memory Output Fields

Field Name	Field Description
PIC	Name of the PIC.
Total IDP data plane memory	Total memory space that is allocated for the IDP data plane. NOTE: IDP requires a minimum of 5 MB of memory for session inspection.
Used	Used memory space in the data plane.
Available	Available memory space in the data plane.

Sample Output

show security idp memory

```
user@host> show security idp memory
    IDP data plane memory statistics:
                          PIC : FPC 0 PIC 0:
  Total IDP data plane memory : 196 MB
                         Used : 8 MB ( 8192 KB ) ( 4.08% )
                    Available : 188 MB ( 192512 KB ) (95.91%)
```

show security idp policies

Supported Platforms SRX Series, vSRX

Syntax show security idp policies

Release Information Command introduced in Junos OS Release 10.1.

Description Display the list of currently installed policies.

Required Privilege Level view

Related Documentation

- show security idp active-policy on page 546

Output Fields user@host> **show security idp policies**

Sample Output

```
Subscriber:  s0,       Installed policies:  1

  ID     Name    Sessions    Memory    detector

  0      new1      0         10179       9.2.160090324
```

show security idp policy-commit-status

Supported Platforms SRX Series, vSRX

Syntax show security idp policy-commit-status

Release Information Command introduced in JUNOS OS Release 10.4.
Starting with Junos OS Release 12.3X48-D15 and Junos OS Release 17.3R1, a new pattern matching engine is introduced for the SRX Series IDP feature. This scanning mechanism helps improve performance and policy loading. The new engine is 9.223 times faster than the existing DFA engine.

Description Display the IDP policy commit status. For example, status of policy compilation or load.

Required Privilege Level view

Related Documentation

- show security idp status on page 594
- show security idp policy-commit-status clear on page 586

List of Sample Output show security idp policy-commit-status on page 585

Sample Output

show security idp policy-commit-status

```
user@host> show security idp policy-commit-status
IDP policy[/var/db/idpd/bins/test.bin.gz.v] and
detector[/var/db/idpd/sec-repository/installed-detector/libidp-detector.so.tgz.v]
 loaded successfully.

The loaded policy size is:45583070 Bytes
```

show security idp policy-commit-status clear

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp policy-commit-status clear
```

Release Information Command introduced in Junos OS Release 10.4.

Description Clear the IDP policy commit status.

Required Privilege Level clear

Related Documentation

- show security idp policy-commit-status on page 585

Output Fields This command produces no output.

show security idp policy-templates

Supported Platforms SRX Series, vSRX

Syntax show security idp policy-templates

Release Information Command introduced in Junos OS Release 10.1.

Description Display the list of available policy templates.

Required Privilege Level view

Related Documentation

- show security idp active-policy on page 546

Output Fields user@host> show security idp policy-templates

Sample Output

```
DMZ_Services
DNS_Service
File_Server
Getting_Started
IDP_Default
Recommended
Web_Server
```

show security idp predefined-attacks

Supported Platforms SRX Series, vSRX

Syntax

```
show security idp predefined-attacks
filters ( category | severity | direction)
```

Release Information Command introduced in Junos OS Release 10.1.

Description Display information about predefined attacks using optional filters.

Options filters (Optional)

- **category**—Show predefined attacks in different categories.
- **severity**—Show predefined attacks based on different severities.
  - **critical**
  - **info**
  - **major**
  - **minor**
  - **warning**
- **direction** — Show predefined attacks for different directions.
  - **any**
  - **client-to-server**
  - **exclude-any**
  - **exclude-client-to-server**
  - **exclude-server-to-client**
  - **server-to-client**

Required Privilege Level view

Output Fields user@host> **show security idp predefined-attacks filters category APP**

Sample Output

```
APP:AMANDA:AMANDA-ROOT-OF1
APP:AMANDA:AMANDA-ROOT-OF2
APP:ARKEIA:TYPE-77-OF
APP:CA:ALERT-SRV-OF
APP:CA:ARCSRV:TCP-BOF
APP:CA:ARCSRV:UA-OF
```

```
APP:CA:IGATEWAY-BOF
APP:CA:LIC-COMMAND-OF
APP:CA:LIC-GCR-OF
APP:CA:LIC-GETCONFIG-OF
APP:CA:LIC-GETCONFIG-OF2
APP:CA:LIC-PUTOLF-OF
APP:CDE-DTSPCD-OF
APP:DOUBLETAKE
APP:ETHEREAL:DISTCC-OF
APP:HPOVNNM:HPOVTRACE-OF
APP:KERBEROS:GSS-ZERO-TOKEN
APP:KERBEROS:KBR-DOS-TCP-2
APP:MDAEMON:FORM2RAW-OF
APP:MERCURY-BOF
APP:MISC:MCAFFEE-SRV-HDR
APP:NTOP-WEB-FS1
APP:PPTP:MICROSOFT-PPTP
APP:REMOTE:TIMBUKTU-AUTH-OF
```

user@host> show security idp security-package predefined-attacks filters category FTP severity critical direction client-to-server

```
FTP:COMMAND:WZ-SITE-EXEC
FTP:DIRECTORY:TILDE-ROOT
FTP:EXPLOIT:OPENFTPD-MSG-FS
FTP:OVERFLOW:OPENBSD-FTPD-GLOB
FTP:OVERFLOW:PATH-LINUX-X86-3
FTP:OVERFLOW:WFTPD-MKD-OVERFLOW
FTP:OVERFLOW:WUBSD-SE-RACE
FTP:PROFTP:OVERFLOW1
FTP:PROFTP:PPC-FS2
FTP:SERVU:CHMOD-OVERFLOW
FTP:SERVU:LIST-OVERFLOW
FTP:SERVU:MDTM-OVERFLOW
FTP:WU-FTP:IREPLY-FS
```

show security idp security-package-version

Supported Platforms SRX Series, vSRX

Syntax show security idp security-package-version

Release Information Command introduced in Junos OS Release 9.2.

Description Display information of the currently installed security package version and detector version.

Required Privilege Level view

Related Documentation

- security-package on page 464
- request security idp security-package download on page 527
- request security idp security-package install on page 530

List of Sample Output show security idp security-package-version on page 590

Output Fields Table 116 on page 590 lists the output fields for the **show security idp security-package-version** command. Output fields are listed in the approximate order in which they appear.

Table 116: show security idp security-package-version Output Fields

Field Name	Field Description
Attack database version	Attack database version number that is currently installed on the system.
Detector version	Detector version number that is currently installed on the system.
Policy template version	Policy template version number that is currently installed on the system.

Sample Output

show security idp security-package-version

```
user@host> show security idp security-package-version
  Attack database version:1154(Mon Apr 28 15:08:42 2008)
  Detector version :9.1.140080400
  Policy template version :7
```

show security idp ssl-inspection key

Supported Platforms SRX Series, vSRX

Syntax **show security idp ssl-inspection key [<*key-name*> [server <*server-ip*>]]**

Release Information Command introduced in Junos OS Release 9.3.

Description Display SSL keys added to the system along with their associated server IP addresses.

Options

- ***key-name*** —(Optional) Name of SSL private key.
- **server *server-ip*** —(Optional) Server IP address associated for specified key.

Required Privilege Level view

List of Sample Output show security idp ssl-inspection key on page 591
show security idp ssl-inspection key key2 on page 592

Output Fields Table 117 on page 591 lists the output fields for the **show security idp ssl-inspection key** command. Output fields are listed in the approximate order in which they appear.

Table 117: show security idp ssl-inspection key Output Fields

Field Name	Field Description
Total SSL keys	Total number of SSL keys.
key	Name of the SSL private key.
server	Server IP address associated with the SSL keys.

Sample Output

show security idp ssl-inspection key

```
user@host> show security idp ssl-inspection key
Total SSL keys : 4

SSL Server key and ip address:

  Key : key1, server : 1.1.0.1
  Key : key1, server : 1.1.0.2
  Key : key2, server : 2.2.0.1
  key : key3
```

Sample Output

show security idp ssl-inspection key key2

```
user@host> show security idp ssl-inspection key key2
SSL Server key and ip address:

  Key : key2, server : 2.2.0.1
```

show security idp ssl-inspection session-id-cache

Supported Platforms SRX Series, vSRX

Syntax show security idp ssl-inspection session-id-cache

Release Information Command introduced in Junos OS Release 9.3.

Description Display all the SSL session IDs in the session ID cache. Each cache entry is 32 bytes long.

Required Privilege Level view

Related Documentation

- clear security idp ssl-inspection session-id-cache on page 525

List of Sample Output show security idp ssl-inspection session-id-cache on page 593

Output Fields Table 118 on page 593 lists the output fields for the **show security idp ssl-inspection session-id-cache** command. Output fields are listed in the approximate order in which they appear.

Table 118: show security idp ssl-inspection session-id-cache Output Fields

Field Name	Field Description
Total SSL session identifiers	Total number of SSL session identifiers stored in the session ID cache.

Sample Output

show security idp ssl-inspection session-id-cache

```
user@host> show security idp ssl-inspection session-id-cache
SSL session identifiers :

c98396c768f983b515d93bb7c421fb6b8ce5c2c5c230b8739b7fcf8ce9c0de4e
a211321a3242233243c3dc0d421fb6b8ce5e4e983b515d932c5c230b87392c

Total SSL session identifiers : 2
```

show security idp status

Supported Platforms SRX Series, vSRX

Syntax show security idp status

Release Information Command introduced in Junos OS Release 9.2. Multiple detector information introduced in Junos OS Release 10.1. Output changed to support IDP dedicated mode in Junos OS Release 11.2.

Description Display the status of the current IDP policy.

Required Privilege Level view

List of Sample Output show security idp status on page 595

Output Fields Table 119 on page 594 lists the output fields for the **show security idp status** command. Output fields are listed in the approximate order in which they appear.

Table 119: show security idp status Output Fields

Field Name	Field Description
State of IDP	Status of current IDP policy.
Packets/second	The aggregated throughput (packets per second) for the system.
KBits/second	The aggregated throughput (kilobits per second) for the system.
Latency	• min—Minimum delay for a packet to receive and return by a node in microseconds. • max—Maximum delay for a packet to receive and return by a node in microseconds. • ave—Average delay for a packet to receive and return by a node in microseconds.
Packet Statistics	Statistics for ICMP, TCP, and UDP packets.
Flow Statistics	Flow-related system statistics for ICMP, TCP, and UDP packets.
Session Statistics	Session-related system statistics for ICMP, TCP, and UDP packets.
Number of SSL Sessions	Number of current SSL sessions.
Policy Name	Name of the running policy. If IDP is configured for logical systems, **idp-policy-combined** is displayed.
Running Detector Version	Current version of the running detector.
Forwarding process mode	IDP dedicated mode: **default, equal, idp,** or **firewall**.

Sample Output

show security idp status

```
user@host> show security idp status
State of IDP: 2-default,  Up since: 2010-02-04 13:37:16 UTC (17:15:02 ago)

Packets/second: 5               Peak: 11 @ 2010-02-05 06:51:58 UTC
KBits/second  : 2               Peak: 5 @ 2010-02-05 06:52:06 UTC
Latency (microseconds): [min: 0] [max: 0] [avg: 0]

Packet Statistics:
[ICMP: 0] [TCP: 82] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]
  TCP: [Current: 2] [Max: 6 @ 2010-02-05 06:52:08 UTC]
  UDP: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]
  Other: [Current: 0] [Max: 0 @ 2010-02-05 06:49:51 UTC]

Session Statistics:
[ICMP: 0] [TCP: 1] [UDP: 0] [Other: 0]

Policy Name : sample
Running Detector Version : 10.4.160091104
```

show security idp status detail

Supported Platforms SRX Series, vSRX

Syntax show security idp status detail

Release Information Command introduced in Junos OS Release 10.1. Output changed to support IDP dedicated mode in Junos OS Release 11.2.

Description Display statistics for each Services Processing Unit (SPU), including multiple detector information for each SPU.

Required Privilege Level view

Sample Output

show security idp status detail

```
user@host> show security idp status detail
  PIC : FPC 1 PIC 1:
State of IDP: Default,  Up since: 2011-03-29 17:25:07 UTC (00:02:48 ago)

Packets/second: 0               Peak: 0 @ 2011-03-29 17:25:07 UTC
KBits/second  : 0               Peak: 0 @ 2011-03-29 17:25:07 UTC
Latency (microseconds): [min: 0] [max: 0] [avg: 0]


Packet Statistics:
[ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:07 UTC]
  TCP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:07 UTC]
  UDP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:07 UTC]
  Other: [Current: 0] [Max: 0 @ 2011-03-29 17:25:07 UTC]

Session Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Number of SSL Sessions : 0

 PIC : FPC 1 PIC 0:

State of IDP: Default,  Up since: 2011-03-29 17:25:08 UTC (00:02:47 ago)

Packets/second: 0               Peak: 0 @ 2011-03-29 17:25:08 UTC
KBits/second  : 0               Peak: 0 @ 2011-03-29 17:25:08 UTC
Latency (microseconds): [min: 0] [max: 0] [avg: 0]

Packet Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:08 UTC]
  TCP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:08 UTC]
  UDP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:08 UTC]
  Other: [Current: 0] [Max: 0 @ 2011-03-29 17:25:08 UTC]

Session Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Number of SSL Sessions : 0

 PIC : FPC 0 PIC 1:

State of IDP: Default,  Up since: 2011-03-29 17:25:04 UTC (00:02:51 ago)


Packets/second: 0               Peak: 0 @ 2011-03-29 17:25:04 UTC
KBits/second  : 0               Peak: 0 @ 2011-03-29 17:25:04 UTC
Latency (microseconds): [min: 0] [max: 0] [avg: 0]

Packet Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]

Flow Statistics:
  ICMP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:04 UTC]
  TCP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:04 UTC]
  UDP: [Current: 0] [Max: 0 @ 2011-03-29 17:25:04 UTC]
  Other: [Current: 0] [Max: 0 @ 2011-03-29 17:25:04 UTC]

Session Statistics:
 [ICMP: 0] [TCP: 0] [UDP: 0] [Other: 0]


Number of SSL Sessions : 0

  PIC : FPC 1 PIC 1:

  Policy Name : none

  PIC : FPC 1 PIC 0:

  Policy Name : none

  PIC : FPC 0 PIC 1:

  Policy Name : none

Forwarding process mode : maximizing sessions  firewall
```