CONFIGURATION GUIDE

Extreme SLX-OS Monitoring Configuration Guide, 18s.1.03

Supporting the ExtremeSwitching SLX 9140 and SLX 9240 Switches

9036275-00

Legal Notice

Extreme Networks, Inc. reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information on Extreme Networks trademarks, please see: www.extremenetworks.com/company/legal/trademarks

Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses. End-user license agreements and open source declarations can be found at: www.extremenetworks.com/support/policies/software-licensing

Contents

Preface

This section discusses the conventions used in this guide, ways to provide feedback, additional help, and other Extreme Networks® publications.

Conventions

This section discusses the conventions used in this guide.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used to highlight specific words or phrases.

Format	Description
bold text	Identifies command names.
	Identifies keywords and operands.
	Identifies the names of GUI elements.
	Identifies text to enter in the GUI.
italic text	Identifies emphasis.
	Identifies variables.
	Identifies document titles.

Format	Description
`Courier font`	Identifies CLI output. Identifies command syntax examples.

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Documentation and Training

To find Extreme Networks product guides, visit our documentation pages at:

Current Product Documentation	www.extremenetworks.com/documentation/
Archived Documentation (for earlier versions and legacy products)	www.extremenetworks.com/support/documentation-archives/
Release Notes	www.extremenetworks.com/support/release-notes
Hardware/Software Compatibility Matrices	https://www.extremenetworks.com/support/compatibility-matrices/
White papers, data sheets, case studies, and other product resources	https://www.extremenetworks.com/resources/

Training

Extreme Networks offers product training courses, both online and in person, as well as specialized certifications. For more information, visit www.extremenetworks.com/education/.

Getting Help

If you require assistance, contact Extreme Networks using one of the following methods:

Extreme Portal	Search the GTAC (Global Technical Assistance Center) knowledge base, manage support cases and service contracts, download software, and obtain product licensing, training, and certifications.
The Hub	A forum for Extreme Networks customers to connect with one another, answer questions, and share ideas and feedback. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
Call GTAC	For immediate support: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: www.extremenetworks.com/support/contact

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

Subscribing to Service Notifications

You can subscribe to email notifications for product and software release announcements, Vulnerability Notices, and Service Notifications.

1. Go to www.extremenetworks.com/support/service-notification-form.
2. Complete the form with your information (all fields are required).
3. Select the products for which you would like to receive notifications.

   NOTE
   You can modify your product selections or unsubscribe at any time.

4. Click **Submit**.

Providing Feedback to Us

Quality is our first concern at Extreme Networks, and we have made every effort to ensure the accuracy and completeness of this document. We are always striving to improve our documentation and help you work better, so we want to hear from you! We welcome all feedback but especially want to know about:

- Content errors or confusing or conflicting information.
- Ideas for improvements to our documentation so you can find the information you need faster.
- Broken links or usability issues.

If you would like to provide feedback to the Extreme Networks Information Development team, you can do so in two ways:

- Use our short online feedback form at https://www.extremenetworks.com/documentation-feedback/.
- Email us at documentation@extremenetworks.com.

Please provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

About This Document

Supported hardware and software

In those instances in which procedures or parts of procedures documented here apply to some devices but not to others, this guide identifies exactly which devices are supported and which are not.

Although many different software and hardware configurations are tested and supported by Extreme Networks, Inc. for this SLX-OS release, documenting all possible configurations and scenarios is beyond the scope of this document.

The following hardware platforms are supported by this release:

- ExtremeSwitching SLX 9140
- ExtremeSwitching SLX 9240

NOTE
Some of the commands in this document use a slot/port designation. Because the SLX 9140 and the SLX 9240 do not contain line cards, the slot designation must always be "0" (for example, 0/1 for port 1).

What's new in this document

The following table includes descriptions of new information for this SLX-OS software release.

TABLE 1 Summary of enhancements in this SLX-OS release

Feature	Description	Described in
Application Telemetry	Application Telemetry is a tool used in Extreme Analytics that uses sFlow to extract network analytics information, such as running application details, flow pathways, latency, bandwidth, and so on.	Application Telemetry on page 57

For complete information, refer to the *Release Notes*.

Port Mirroring

Port mirroring overview

Port mirroring, known as Switched Port Analyzer (SPAN) is used on a network switch to send a copy of network packets seen on one switch port to a network monitoring connection on another switch port.

Unlike a hub which broadcasts any incoming traffic to all ports, a switch acts more intelligently and forwards traffic accordingly. If the user is interested in listening or snooping on traffic that passes through a particular port, port mirroring is necessary to artificially copy the packets to a port connected to the analyzer.

SPAN guidelines and limitations

Consider the following topics when configuring SPAN.

Standard SPAN guidelines and limitations

NOTE
A SPAN session consists of either a single egress port, a single ingress port, or both.

We recommend that you be aware of the following additional standard guidelines for and limitations of SPAN connections:

- The mirror port should not be configured to carry normal traffic.
- The destination mirror port can handle up to 100 Gbps (line rate) worth of mirror traffic, depending on the capability of the destination port. If multiple ports, or both flows on the same port, are mirrored to the same destination mirror port, then only the destination port's capacity worth of mirror traffic is mirrored and the remaining traffic is ignored.
- If the source port receives burst traffic and the destination mirror port cannot handle all the bursts, some of the burst traffic is not mirrored.
- Mirroring of Inter-Chassis Link (ICL) is supported.
- Mirroring of LAG or port-channel interfaces is not supported, but LAG members can be mirrored.
- Ethernet Pause frames are not mirrored.
- The multicast and broadcast statistics are correctly updated on TX ports for mirrored traffic.
- All commands except for **shutdown** and **no shutdown** are blocked on a destination mirror port.
- The interface counters are cleared when a port is successfully designated as a destination mirror port.
- The **show interface** command hides the Receive Statistics and Rate Info (Input) information for a destination mirror port.
- The MTU of a port should be set to the default value before it is made a destination mirror port. When the port is successfully designated as the destination mirror, the MTU of that port is automatically set to the maximum value of 9216 bytes. When the port becomes a non-destination mirror, the MTU is restored to the default value.
- Port mirroring is supported on any physical front-end user-configurable port. The source port can be part of a LAG, , VLAN, or any other user configuration

- A maximum of 512 mirror sessions are supported. A mirror session consists of either a single egress port, a single ingress port, or both.

  NOTE
  Remote Span (RSPAN) is not supported on this platform.

Configuring SPAN

Refer also to Standard SPAN guidelines and limitations on page 13.

Configuring ingress SPAN

To configure SPAN for incoming packets only, do the following:

1. Open a monitor session and assign a session number.

   ```
   switch(config)# monitor session 1
   ```

2. Configure the source port and the destination port, with the **rx** parameter for received packets.

   The destination port is always an external port.

   ```
   switch(config-session-1)# source ethernet 0/1 destination ethernet 0/2 direction rx
   ```

   NOTE
   If the following error is displayed, use the **lldp disable** or related configuration command in interface subtype configuration mode to disable LLDP or other related configuration on the destination port before preceding: `% Error: Destination port cannot have L2/L3/Qos/ACL/802.1x/LAG/LLDP/Port-profile/MTU/StormControl/RMON configuration on it..`

3. Optional: Use the **description** command to add a label to the monitor session.

   ```
   switch(config-session-1)# description Hello World!
   ```

4. Optional: Repeat steps 1 and 2 as needed for additional ports.

   A monitor session can have only one source port. For additional ports you must create additional monitor sessions as needed for additional port mirroring sessions.

Configuring egress SPAN

To configure SPAN for outgoing packets only, do the following.

1. Open a monitor session and assign a session number

   ```
   switch(config)# monitor session 1
   ```

2. Configure the source port and the destination port, with the **tx** parameter for transmitted packets.

   The destination port is always an external port.

   ```
   switch(config-session-1)# source ethernet 0/1 destination ethernet 0/2 direction tx
   ```

   NOTE
   If the following error is displayed, use the interface **lldp disable** or related configuration command to disable LLDP or other related configuration on the destination port before preceding: `% Error: Destination port cannot have L2/L3/Qos/ACL/802.1x/LAG/LLDP/Port-profile/MTU/StormControl/RMON configuration on it..`

3. Optional: Use the **description** command to add a label to the monitor session.

   ```
   switch(config-session-1)# description Hello World!
   ```

4. Optional: Repeat steps 1 and 2 as needed for additional ports.

   A monitor session can have only one source port. For additional ports you must create additional monitor sessions as needed for additional port mirroring sessions.

Configuring bidirectional SPAN

To configure SPAN for packets traveling in both directions, do the following.

1. Open a monitor session and assign a session number

   ```
   switch(config)# monitor session 1
   ```

2. Configure the source port and the destination port, with the **both** parameter for all packets.

   The destination port is always an external port.

   ```
   switch(config-session-1)# source ethernet 0/1 destination ethernet 0/2 direction both
   ```

   NOTE
   One of the following error messages may appear. If so, use the interface **lldp disable** command to disable LLDP on the destination port before preceding.

   - % Error: Destination port cannot have LLDP configuration on it.
   - % Error: Destination port cannot have L2/L3/Qos/ACL/802.1x/LAG/LLDP/Port-profile/MTU/StormControl/RMON configuration on it.

3. Optional: Use the **description** command to add a label to the monitor session.

   ```
   switch(config-session-1)# description Hello World!
   ```

4. Optional: Repeat steps 1 and 2 as needed for additional ports.

   A monitor session can have only one source port. For additional ports you must create additional monitor sessions as needed for additional port mirroring sessions.

Deleting a SPAN connection from a session

To remove a single connection from a SPAN session, do the following.

1. Display the existing configuration of the monitor session.

```
switch# show monitor session 1
```

2. Open an existing monitor session.

```
switch(config)# monitor session 1
```

3. Use the **no** keyword to delete a particular port connection.

```
switch(config-session-1)# no source ethernet 0/1 destination ethernet 0/2 direction both
```

4. Display the monitor session again to confirm the deletion of the connection.

```
switch# show monitor session 1
```

Deleting a SPAN session

To remove a SPAN session, do the following:

1. Display the existing configuration of the monitor session.

```
switch# show monitor session 1
```

2. Delete the existing monitor session by using the **no monitor session** command.

```
switch(config)# no monitor session 1
```

3. Return to Privileged EXEC mode with the **exit** command.
4. Display the monitor session again to confirm the deletion of the connection.

```
switch# show monitor session 1
```

Flow-based SPAN

You can snoop on traffic that passes through a particular port, using flow-based SPAN to copy the packets to a port connected to the analyzer.

Flow-based SPAN selectively mirrors the traffic coming on the source port that matches an ACL-based filter to a destination port.

For example, assume there are two streams of traffic, one from the source Mac1 and other from source Mac2 are being forwarded from eth 0/1 eth 0/2. You can, with the help of an ACL to permit only source Mac1 traffic, configure a flow-based SPAN session with the source on port te1/0/1 and port te1/0/2 as the destination port. All traffic coming in on port te1/0/1 originating from source Mac1 will be duplicated and sent to port te1/0/2. No mirroring occurs for traffic originating from source Mac2.

Consider the following guidelines and restrictions for flow-based SPAN:

- Only ingress direction of the service policy is supported with the current infrastructure.
- Port-based SPAN sessions cannot be specified as the SPAN action.
- Deny rules in an service ACL is a pass through in flow-based QoS. Only permit rules with SPAN action result in flow-based SPAN.

- If a rule is configured as permit in flow-based ACL with SPAN action and the same rule is configured as deny in a user policy, the packet is dropped as per the user policy and the same is mirrored to the SPAN destination port.
- In a class map, if the SPAN action co-exists with any other QoS action (such as DSCP marking which results in frame editing), the mirrored packet is the original packet and hence does not reflect the frame editing done, as per the QoS action.

Configuring flow-based SPAN

You can replicate traffic from a defined source and direct it to snooping software on a designated port.

This task assumes you have already completed the following tasks:

- You have already created a policy map instance.
- You have already created a class map for the policy map.

To configure flow-based SPAN, perform the following task in privileged EXEC mode.

1. Enter global configuration mode.

```
device# configure terminal
```

2. Create the monitor session.

```
device(config)# monitor session 1
```

3. Set the destination port for the replicated traffic for SPAN.

```
device(config-session-1)# destination ethernet 0/2
```

4. Activate the pre-defined policy map.

```
device(config)# policy-map policymap
```

5. Activate the pre-defined class for the policy map.

```
device(config-policymap)# class policyclass
```

6. Activate the span session and assign it an identifying number.

```
device(config-policymap-class)# span session 1
```

7. Return to global configuration mode by executing the **exit** command twice.

```
device(config-policymap-class)# exit
```

8. Enter configuration mode for the source interface.

```
device(config)#interface ethernet 0/1
```

9. Bind the policy to the interface.

```
device(conf-if-eth-0/1)# service-policy in policymap
```

10. Confirm the session with the **show monitor** command.

```
device# show monitor

Session                : 1
Description            : [None]
Type                   : Flow Based SPAN
Enabled on the source interfaces
************************************************
Name                  State       Direction
************************************************
Eth 0/1               (Up)        Rx
Destination Interface  : Eth 0/2 (Up)
device#
```

Deleting the flow-based SPAN session

You remove the flow-based SPAN session by disassociating the span session from the policy-map . The pre-defined policy map and class as such are not deleted.

1. Activate the pre-defined policy map.

```
device(config)# policy-map policymap
```

2. Activate the pre-defined class for the policy map.

```
device(config-policymap)# class policyclass
```

3. Deactivate the span session with the **no span session** version of the command and the identifying number.

```
device(config-policymap-class)# no span session 1
```

4. Delete the span session.

```
device(config)# no monitor session 1
```

Network-Elements Telemetry

Network-elements telemetry overview

Network-elements *telemetry* collects data and measurements at regular intervals and transmits them to external equipment for monitoring and analysis.

> NOTE
> For the Network Packet Broker (NPB)-grid telemetry implementation, refer to the *Extreme SLX-OS Network Packet Broker Configuration Guide*.

Telemetry *profiles* are basic elements of the various SLX-OS telemetry implementations. Each profile is designed to monitor a specific grouping of data, for example, queue or interface statistics. For profile descriptions and implementation, refer to Telemetry profiles on page 19.

Telemetry data collected on the network elements is transmitted using two approaches:

- External-collector streaming on page 23
- gRPC-server streaming on page 27

You can stream telemetry concurrently to no more than six collectors and gRPC clients.

Telemetry profiles

Telemetry profiles determine which types of data are collected and parameters that govern the data collection.

Profiles contain the following elements:

- Attributes (usually counters), which you can selectively remove from the profile
- An **interval** value (how often data are sent), which you can modify
- (Most profiles) The interfaces that you want the profile to monitor

The following telemetry profile types and profiles are supported:

interface

Of the **interface** profile-type, the only profile supported is **default_interface_statistics**. This profile tracks data related to the physical interface. You need to specify monitored interfaces and can modify the default streaming interval.

The fields supported by default for this profile are as follows:

- ethernet and port-channel interfaces
- interval
- In/Out packets

- In/Out unicast packets
- In/Out broadcast packets
- In/Out multicast packets
- In/Out packets per second
- In/Out octets
- In/Out errors
- In/Out CRC errors
- In/Out discards
- In/Out bandwidth
- In/Out link-utilization

pbr

Of the **pbr** profile-type, the only profile supported is **default_pbr_statistics**. This profile tracks policy-based routing (PBR) data. You need to specify monitored interfaces and can modify the default streaming interval.

The fields supported by default for this profile are as follows:

- ethernet and port-channel interfaces
- interval
- acl-byte-count
- acl-hit-count
- acl-name
- acl-seq-num
- if-name
- num-rules
- route-map

system-utilization

Of the **system-utilization** profile-type, the only profile supported is **default_system_utilization_statistics**. This profile tracks system-related data. You can modify the default streaming interval.

The fields supported by default for this profile are as follows:

- Interval
- Total system memory
- Total used memory
- Total free memory
- Cached memory
- Buffers
- Total swap memory
- User free memory
- Kernel free memory

- Total swap memory
- Total free swap memory
- Total used swap memory
- User process
- System process
- Niced process
- In/out wait
- HW interrupt
- SW interrupt
- Idle state
- Steal time
- Up time

Queue profiles

Of the **queue** profile-type, the only profile supported is **default_queue_statistics**. This profile tracks data related to the queue. You need to specify monitored interfaces and can modify the default streaming interval.

Of the **enhanced-queue-discard-pkts** profile-type, the only profile supported is **default_enhanced_queue_discard_pkts_statistics**. This profile tracks data related to discarded packets. You need to specify monitored interfaces and can modify the default streaming interval.

The following table summarizes profile support for queue data.

TABLE 2 Profile support for queue data

Profile	Description
default_enhanced_queue_discard_pkts_statistics	Captures summary of 32 queues having the most number of packets discarded, in descending order of packet discards observed. Indexed by Ethernet interface. Data stream contains: • Slot-id • Device-id (~Ifindex) • Queue-id • Discard Packet Counts
default_queue_statistics	Captures all queue statistics per specified Ethernet interface. Data stream contains: • Slot-id • Device-id (~Ifindex) • Queue-id • EnQ Pkt Count • EnQ Byte Count • Discard Pkt Count • Discard Byte Count • Current Queue Size • Max Queue Depth SIze

event

Of the **event** profile-type, the only profile supported is **default_event_statistics**. This profile tracks up/down interface events.

The fields supported by default for this profile are as follows:

- event-class
- payload
- severity
- timestamp

lldp

Of the **lldp** profile-type, the only profile supported is **default_lldp_statistics**. This profile tracks LLDP neighbor information, including link states. You need to specify monitored Ethernet interfaces and can modify the default streaming interval.

The fields supported by default for this profile are as follows:

- if-name
- link-state
- local-chassis-id
- local-lag-id
- local-port
- remote-chassis-id
- remote-lag-id
- remote-port
- speed
- update-type

Configuring telemetry profiles

You configure telemetry profiles by specifying interfaces to monitor, with options to remove one or more attributes and to modify the streaming interval.

1. In privileged EXEC mode, enter **configure terminal**.

```
device# configure terminal
```

2. Enter the **telemetry profile** command to configure the profile.

```
device(config)# telemetry profile interface default_interface_statistics
```

3. (For profiles that require specifying interfaces) Enter one of the **interface** commands to specify monitored interfaces.
   - To specify individual interfaces or ranges of interfaces, use the **add** keyword.

     ```
     device(config-interface-default_interface_statistics)# interface ethernet add 0/1-2,0/7
     ```

   - To specify all interfaces, use the **all** keyword.

     ```
     device(config-interface-default_interface_statistics)# interface ethernet all
     ```

   - To specify all—with the exception of certain—interfaces, use the **except** keyword.

     ```
     device(config-interface-default_interface_statistics)# interface ethernet except 0/1-2,0/7
     ```

   - To remove all—with the exception of certain—interfaces, use the **remove** keyword.

     ```
     device(config-interface-default_interface_statistics)# interface ethernet remove 0/3-4,0/6
     ```

   - To remove all interfaces, use the **none** keyword.

     ```
     device(config-interface-default_interface_statistics)# interface ethernet none
     ```

     NOTE
     The **interface** and **pbr** profile types enable you to specify port-channels for monitoring—with the option of also specifying Ethernet interfaces. For such multiple support, enter multiple **interface** commands.

4. To modify the default interval, enter the **interval** command.

   ```
   device(config-interface-default_interface_statistics)# interval 30
   ```

5. To remove a default attribute, enter the **no add** command.

   ```
   device(config-system-utilization-default_system_utilization_statistics)# no add buffers
   ```

6. To restore a default attribute that was previously removed, enter the **add** command.

   ```
   device(config-system-utilization-default_system_utilization_statistics)# add buffers
   ```

7. To exit configuration mode—saving the configuration—enter **exit**.

   ```
   device(config-interface-default_interface_statistics)# exit
   ```

The following example specifies the monitored interfaces and changes the default interval.

```
device# configure terminal
device(config)# telemetry profile interface default_interface_statistics
device(config-interface-default_interface_statistics)# interval 30
device(config-interface-default_interface_statistics)# interface 0/1-2,0/7
device(config-interface-default_interface_statistics)# exit
```

External-collector streaming

In the external-collector telemetry-streaming implementation, your monitored device streams data to one or more *collector* devices—for monitoring and analysis.

For each target collector device, you configure a local **telemetry collector** object that specifies the following parameters:

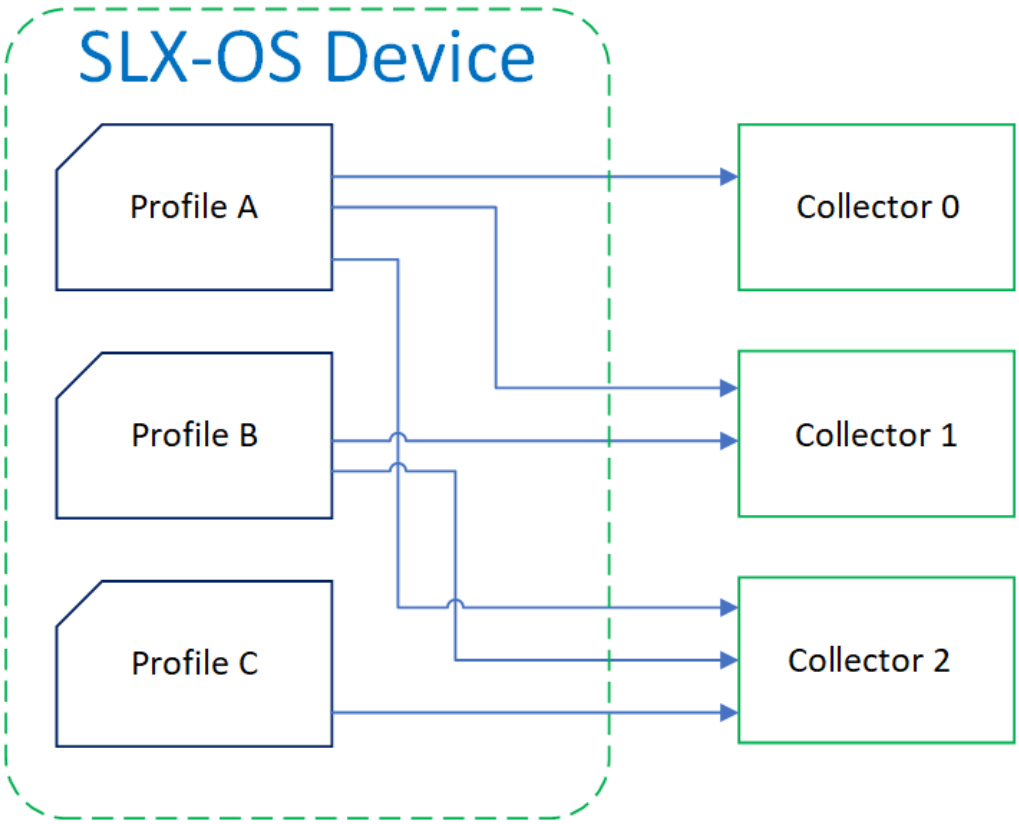
- One or more telemetry profiles
- IPv4 or IPv6 address and port of the collector device

- Encoding format (Google protocol buffers (GPB) or JavaScript object notation (JSON))
- Activation

The following diagram depicts the many-to-many relationships between profiles and collectors:

- You can specify multiple profiles in a collector.
- You can specify a profile in multiple collectors.

FIGURE 1 Profile/collector relationships in external-collector streaming



NOTE
The preceding diagram does not depict local **telemetry collector** objects.

Configuring telemetry collectors

This task configures telemetry streaming to an external collector—for monitoring and analysis.

1. In privileged EXEC mode, enter **configure terminal**.

```
device# configure terminal
```

2. Enter the **telemetry collector** command, specifying a collector name.

```
device(config)# telemetry collector collector_1
```

3. Enter the **ip port** command, specifying the IPv4 or IPv6 address and port of the collector.

```
device(config-collector-collector_1#) ip 10.168.112.10 port 1
```

4. Enter the **profile** command to add one or more telemetry profiles.

```
device(config-collector-collector_1)# profile system-profile  default_system_utilization_statistics
device(config-collector-collector_1)# profile interface-profile default_interface_statistics
```

5. Enter the **encoding** command to specify the encoding format.

```
device(config-collector-collector_1)# encoding ?
Possible completions:
  gpb    Google protobuf encoding
  json   JSON encoding
device(config-collector-collector_1)# encoding json
```

6. Enter the **activate** command to activate the collector.

```
device(config-collector-collector_1)# activate
```

7. To confirm the configuration, enter the **show running-configuration telemetry collector** command.

```
device(config-collector-collector_1)# do show running-configuration telemetry collector
telemetry collector <collector-profile-1>
  ip <ipv4address1> port <portNum>
  profile system-utilization default_system_utilization_statistics
  profile interface default_interface_statistics
  use-vrf mgmt-vrf
  encoding json
  activate
```

8. To display the status of a telemetry collector, enter the **show telemetry collector** command.

```
device(config-collector-collector_1)# do show telemetry collector name collector_1
Telemetry data is streamed to collector_1 on 10.128.116.10 and port 1, with transport as tcp.

Profiles Streamed                     Interval   Uptime      Last Streamed
------------------                    ---------  ----------  ---------------
default_interface_statistics          120 sec    05/10:23    2017-01-15: :05:07:33
default_system_utilization_statistics 300 sec    05/10:23    2017-01-15: :05:07:33
!
```

9. To display the status of active telemetry collector sessions, enter the **show telemetry collector summary** command.

```
device# show telemetry collector summary
Activated Collectors:
--------------------
Name              IP Address:Port       Streaming/Connection Status
-------------     --------------------  --------------------------
 coll1            10.24.65.182:9000     connection_failed
```

The following example configures and activates a telemetry collector.

```
device# configure terminal
device(config)# telemetry collector collector_1
device(config-collector-collector_1)# profile system-profile default_system_utilization_statistics
device(config-collector-collector_1)# profile interface-profile default_interface_statistics
device(config-collector-collector_1)# ip 10.168.112.10 port 1
device(config-collector-collector_1)# encoding json
device(config-collector-collector_1)# activate
```

Configuring third party analytic software

This feature integrates SLX-OS streaming support with third party analytics applications and tools.

SLX-OS supports periodic streaming of operational data; such as interface statistic counters, metrics like memory utilization, processor utilization and important parameters related to protocols like OSPF, BGP and so on. This data can be leveraged to build analytics-based applications that help in gathering and processing the data and provide visibility into network monitoring from different aspects, such as utilization, performance, security, and so on.

Below are the functionalities for this feature:

- Provides basic infrastructure support to gather data from SLX-OS
- Provides sample reports and dashboards built from the data collected

These functionalities are available using a Splunk app called ‘‘Extreme Streaming app for Splunk”, that has the necessary components to extend the functionality of the Splunk platform. The functionalities are:

- Manages the ‘telemetry collector’ to collect the data streamed from SLX-OS box
- Capable of receiving data in both JSON and GPB encoding
- Generates reports and dashboards based on the streamed data
- Multiple network elements can stream data to Splunk in parallel
- Splunk reports and dashboards can be viewed for each network element

The ‘‘Extreme Streaming app for Splunk” must be installed into the required Splunk instance. For complete information, refer to the Splunk company documentation and website at https://www.splunk.com.

The Splunk app listens to port numbers 54326 for ‘JSON encoded’ and 54327 for ‘GPB encoded’ streaming data respectively. User can change this by updating the configuration file as mentioned in “App receiver ports configuration” section of the Splunk application. It is sufficient to have either of the above collector configurations. Having both of them for the same IP address is redundant.

1. Enter global configuration mode.

```
device# configure terminal
```

2. Enter telemetry collector configuration mode for the JSON collector with the **telemetry collector** command .

```
device(config)# telemetry collector SplunkJSONCollector
device(config-telemetry-collector_SplunkJSONCollector)#
```

3. Set the IP address and IPv4 port for the telemetry collector with the **ip** command.

```
device(config-telemetry-collector_SplunkJSONCollector)# ip 10.168.72.116 port 54326
```

4. Exit telemetry collector configuration mode with the **exit** command.

```
device(config-telemetry-collector_SplunkJSONCollector)# exit
device(config)#
```

5. Enter telemetry collector configuration mode for the GPB collector with the **telemetry collector** command .

```
device(config)# telemetry collector SplunkGPBCollector
device(config-telemetry-collector_SplunkJSONCollector)#
```

6. Set the IP address and IPv4 port for the telemetry collector with the **ip** command.

```
device(config-telemetry-collector_SplunkJSONCollector)# ip 10.168.72.117 port 54327
```

gRPC-server streaming

In this telemetry implementation, your monitored device functions as a Google remote procedure call (gRPC) server, streaming data in response to calls from gRPC clients.

The telemetry profiles currently configured with interfaces determine the monitored attributes and the streaming intervals. For details, refer to Telemetry profiles on page 19. Client RPCs must be crafted to request streaming per telemetry profile.

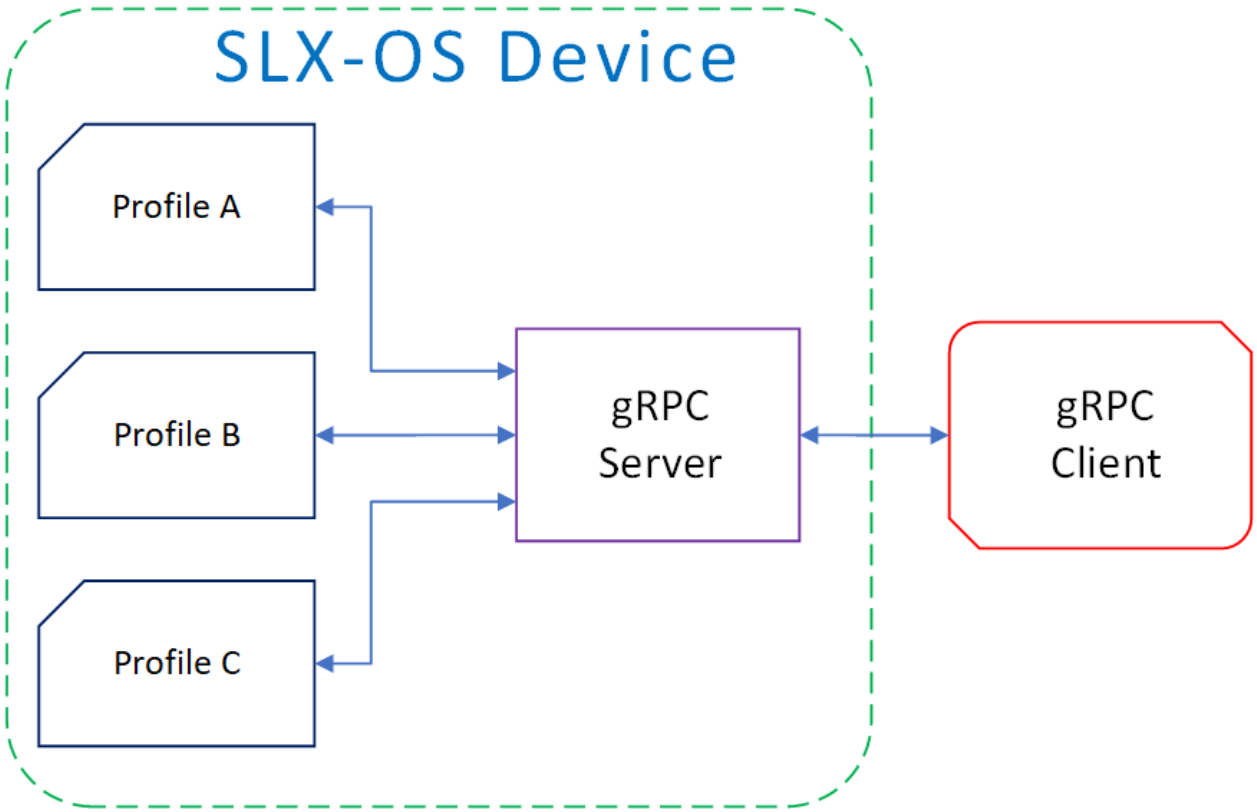
From telemetry-server configuration mode, you can modify the following default settings:

- Port: 50051
- Transport protocol: TCP, which can be modified to SSL.

From telemetry-server configuration mode, you also activate and deactivate the gRPC server.

The following diagram illustrates the gRPC-server telemetry implementation:

FIGURE 2 gRPC-server streaming



Configuring the gRPC telemetry server

This task configures and activates the device gRPC telemetry server.

1. In privileged EXEC mode, enter **configure terminal**.

```
device# configure terminal
```

2. Enter **telemetry server** to configure the gRPC telemetry server.

```
device(config)# telemetry server
```

3. To specify a port other than the default 50051, enter the **port** command.

```
device(config-server-mgmt-vrf)# port 50000
```

4. Enter **activate**.

```
device(config-server-mgmt-vrf)# activate
```

5. Verify the telemetry server status with the **do show telemetry server status** command. The active sessions displayed are initiated by gRPC clients with associated telemetry profiles.

```
device(config-server-mgmt-vrf)# do show telemetry server status

Telemetry Server running on port 50051, with VRF mgmt-vrf and transport as tcp.

Active Sessions:
----------------
Client            Profiles Streamed                    Interval   Uptime      Last Streamed
------            -----------------                    ---------  ----------  ---------------
ClientIP1/Host1   default_interface_statistics          120 sec    05/10:23    2017-01-15: :05:07:33
                  default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33

ClientIP2/Host2   default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33
```

6. Verify the telemetry server configuration with the **do show running-configuration telemetry server** command.

```
device(config-server-mgmt-vrf)# do show running-configuration telemetry server

telemetry server
  transport tcp
  port 50051
  activate
!
```

The following is a complete telemetry server configuration example.

```
device# configure terminal
device(config)# telemetry server
device(config-server-mgmt-vrf)# activate
device(config-server-mgmt-vrf)# do show telemetry server status

Telemetry Server running on port 50051, with VRF mgmt-vrf and transport as tcp.

Active Sessions:
----------------
Client            Profiles Streamed                    Interval   Uptime      Last Streamed
------            -----------------                    ---------  ----------  ---------------
ClientIP1/Host1   default_interface_statistics          120 sec    05/10:23    2017-01-15: :05:07:33
                  default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33

ClientIP2/Host2   default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33
device(config-server-mgmt-vrf)# do how running-configuration telemetry server

telemetry server
  transport tcp
  port 50051
  activate
!
```

Configuring SSL on the gRPC telemetry server

The gRPC-server telemetry implementation supports secure monitoring through SSL transport security.

1. In privileged EXEC mode, enter **configure terminal**.

```
device# configure terminal
```

2. Enter **telemetry server** to configure the gRPC telemetry server.

```
device(config)# telemetry server
```

3. Enter the **activate** command.

```
device(config-server-mgmt-vrf)# activate
```

4. Enter **do telemetry client-cert generate** to generate SSL certificates for the server and client.

```
device(config-server-mgmt-vrf)# do telemetry client-cert generate
```

5. Verify the certificate is active with the **do show telemetry client-cert** command.

   This output displays the SSL public CA certificate that is used for secure connections on the client side for establishing SSL connections, such as streaming with recipients for gRPC clients or destinations.

```
device(config-server-mgmt-vrf)# do show telemetry client-cert

-----BEGIN CERTIFICATE-----
MIIC2jCCAcICAQEwDQYJKoZIhvcNAQEFBQAwMzELMAkGA1UEBhMCQ0ExEDAOBgNV
BAoMB0Jyb2NhZGUxEjAQBgNVBAMMCWxvY2FsaG9zdDAeFw0xNzAzMjExNzQ1NDNa
Fw0xODAzMjExNzQ1NDNaMDMxCzAJBgNVBAYTAkNBMRAwDgYDVQQKDAdCcm9jYWRl
MRIwEAYDVQQDDAlsb2NhbGhvc3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC+YG/CkiNm/BO+u1mYLKP8cpz/0O9CE+fusO0spXxjKfjPAvK7kiogxABm
bg9MQeWl4SbFa5x3q5uyZJxApJ+tAnnWZa+cbj5pmNsQFfIbFOwSAmFyhh/NIp7Y
/wApskKjnVsMFkarqX8W2xKxZreapZFMa9DGpOeh8Jo2yvcTAimFfSJ4nyKlCr1C
DuuaTSvAttC8Z9mEqD9TOaSYwQI0pnfVO+ySgY8ndqDXYdRvl+bV1taghlKOgxMY
J78lyZxYf6CIn22BAaz/f9a5ffS13Hh5Cmurj2dUmmqDE49p2KEVtXQ3D6nuopli
V49ok+z93/40Uq4OVJZJk5Kx8ZuxAgMBAAEwDQYJKoZIhvcNAQEFBQADggEBAI1d
1VkMH9i3SorPIHpbVqbeDe7LPdaFmrT0COr3AFUECw3gBj1Zy82Kp8XkIJJdVCu8
MNm3wTARqeNBY2c3luw6QeA6l4qRIVM4FqNj6rvtqtNZQ9EEKRRwAm0GSVp+uSvu
E88XSXO+r6N+SXQemRIyhNQ7LJq+cDEaP5WfNtKg+zj085Xd0qiB94BKFt5Q+xAa
B7lwuUvT7Yt92aUVXIaZ6aY5oMv4t7+lPBBKjg8cNeywDa9h3yVZYIzSggghu0qu
GZO57qUh5agxqKiEVf9Ya325u5gj73UJsKOSsyVA1HB8RsPEEdz8j8FBAqMNSTQj
8UDtUGpYiYlzyiBUELc=
-----END CERTIFICATE-----
```

6. Enter **transport ssl**.

```
device(config-server-mgmt-vrf)# transport ssl
```

The following example is a complete SSL configuration example.

```
device# configure terminal
device(config)# telemetry server
device(config-server-mgmt-vrf)# activate
device(config-server-mgmt-vrf)# do show telemetry server status

Telemetry Server running on IP 10.128.116.10 and port 1, with transport as tcp.

Active Sessions:
----------------
Client            Profiles Streamed                     Interval   Uptime      Last Streamed
------            -----------------                     ---------  ----------  ---------------
ClientIP1/Host1   default_interface_statistics          120 sec    05/10:23    2017-01-15: :05:07:33
                  default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33

ClientIP2/Host2   default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33

device(config-server-mgmt-vrf)# do telemetry client-cert generate
device(config-server-mgmt-vrf)# do show telemetry client-cert

-----BEGIN CERTIFICATE-----
MIIC2jCCAcICAQEwDQYJKoZIhvcNAQEFBQAwMzELMAkGA1UEBhMCQ0ExEDAOBgNV
BAoMB0Jyb2NhZGUxEjAQBgNVBAMMCWxvY2FsaG9zdDAeFw0xNzAzMjExNzQ1NDNa
Fw0xODAzMjExNzQ1NDNaMDMxCzAJBgNVBAYTAkNBMRAwDgYDVQQKDAdCcm9jYWRl
MRIwEAYDVQQDDAlsb2NhbGhvc3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC+YG/CkiNm/BO+u1mYLKP8cpz/0O9CE+fusO0spXxjKfjPAvK7kiogxABm
bg9MQeWl4SbFa5x3q5uyZJxApJ+tAnnWZa+cbj5pmNsQFfIbFOwSAmFyhh/NIp7Y
/wApskKjnVsMFkarqX8W2xKxZreapZFMa9DGpOeh8Jo2yvcTAimFfSJ4nyKlCr1C
DuuaTSvAttC8Z9mEqD9TOaSYwQI0pnfVO+ySgY8ndqDXYdRvl+bV1taghlKOgxMY
J78lyZxYf6CIn22BAaz/f9a5ffS13Hh5Cmurj2dUmmqDE49p2KEVtXQ3D6nuopli
V49ok+z93/40Uq4OVJZJk5Kx8ZuxAgMBAAEwDQYJKoZIhvcNAQEFBQADggEBAI1d
1VkMH9i3SorPIHpbVqbeDe7LPdaFmrT0COr3AFUECw3gBj1Zy82Kp8XkIJJdVCu8
MNm3wTARqeNBY2c3luw6QeA6l4qRIVM4FqNj6rvtqtNZQ9EEKRRwAm0GSVp+uSvu
E88XSXO+r6N+SXQemRIyhNQ7LJq+cDEaP5WfNtKg+zj085Xd0qiB94BKFt5Q+xAa
B7lwuUvT7Yt92aUVXIaZ6aY5oMv4t7+lPBBKjg8cNeywDa9h3yVZYIzSggghu0qu
GZO57qUh5agxqKiEVf9Ya325u5gj73UJsKOSsyVA1HB8RsPEEdz8j8FBAqMNSTQj
8UDtUGpYiYlzyiBUELc=
-----END CERTIFICATE-----
device(config-server-mgmt-vrf)# transport ssl
device(config-server-mgmt-vrf)# do show telemetry server status

Telemetry Server running on IP 10.128.116.10 and port 1, with transport as ssl.

Active Sessions:
----------------
Client            Profiles Streamed                     Interval   Uptime      Last Streamed
------            -----------------                     ---------  ----------  ---------------
ClientIP1/Host1   default_interface_statistics          120 sec    05/10:23    2017-01-15: :05:07:33
                  default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33

ClientIP2/Host2   default_system_utilization_statistics  300 sec    05/10:23    2017-01-15: :05:07:33
```

Hardware Monitoring

Hardware monitoring overview

Hardware monitoring allows you to monitor CPU and memory usage of the system, interface and optic environmental status, and security status and be alerted when configured thresholds are exceeded.

Policies can be created with default options or custom options for non-default thresholds. When the policies are applied, you can toggle between default settings and saved custom configuration settings and apply actions and thresholds separately. For example, you can choose to use default threshold settings together with a customized subset of available actions, or you can modify some of the threshold settings and use the default action settings. You can also pause monitoring and actions.

System Resource Monitoring (SRM)

The System Resource Monitoring (SRM) provides periodic, continuous check on system-wide memory and per-process memory usages in an active running switch and provide warnings to users regarding abnormally high memory usage.

This helps you to take adequate actions before the system reaching fatal state. This automated information gathering helps to identify those processes which are involved in high memory usage and assist in debugging memory leakage. Based on this information, you can amend configurations to avoid pushing the resource usage over the limit. SupportSave data is also collected so that the root cause of the issue can be analyzed offline and fixed.

With the per-process memory monitoring service enabled, if the high memory usage threshold is crossed for any of the processes, an **alarm** message is generated. If memory usage still goes up to another threshold, a **critical** message is generated. Based on the information available, the resolution has to be worked out manually.

If the system memory monitoring service is enabled, SRM generates raslog to notify that the system memory usage crossed the set threshold. If the CPU utilization monitoring service is enabled, SRM generate raslog to notify that the CPU usage exceeded threshold of 90%.

This functionality is provided by the **resource-monitor** command.

```
device(config)# resource-monitor cpu
Possible completions:
  action      Action to take when CPU usage exceeds threshold
  enable      Enable monitoring CPU usage
  threshold   Threshold for high CPU usage
device(config)# resource-monitor cpu enable
Possible completions:
  action      Action to take when CPU usage exceeds threshold
  threshold   Threshold for high CPU usage
  <cr>
device(config)# resource-monitor cpu enable action raslog threshold
Possible completions:
  <PERCENT:70-100 percent CPU being busy>[90]
device(config)# resource-monitor cpu enable action raslog threshold
```

For more information on commands, please refer the *SLX-OS Command Reference* guide for the SLX 9140 and 9240 Switches.

Configuring system resource monitoring

Execute the following steps to configure resource monitoring.

1. Enter global configuration mode.

```
device# configure terminal
```

2. Issue the **resource-monitor cpu enable** command to enable the CPU utilization monitoring service.

```
device(config)# resource-monitor cpu enable
```

3. Issue the **resource-monitor memory** command to enable the system memory monitoring and generate raslog when the memory usage exceeds threshold value.

```
device(config)# resource-monitor memory enable action raslog threshold 80
```

4. Issue the **resource-monitor process memory** command to enable the per-process memory monitoring and generate alarm or raslog when the usage exceeds alarms threshold or critical threshold respectively.

```
device(config)# resource-monitor process memory  enable alarm 800 critical 1000
```

5. (Optional) Issue the how running-configuration command to view the resource monitoring running configuration.

```
device# show running-config resource-monitor
 resource-monitor cpu enable
 resource-monitor memory enable threshold 80 action raslog
 resource-monitor process memory enable alarm 800 critical 1000
!
device#
```

CPU, memory, and buffer monitoring

SRM allow users to enable or disable the CPU and memory monitoring services and set threshold values. By default, these services are enabled.

With CPU monitoring service enabled, if the system CPU is busy and reaches certain threshold, a RASLOG message is generated. This is informational only and no user action is required.

With the system memory monitoring service enabled, if the system memory is running low and goes below the threshold in the system, SRM notifies the user via RASLOG message. This RASLOG message allows the user to investigate any potential system memory issues easily.

With the process memory monitoring service enabled, if any of the process consumes too much memory and the usage reaches certain threshold, a WARNING message is generated. If the usage still goes up to another threshold, a CRITICAL message will be generated. These RASLOG messages allow the user to identify any potential process resource issues.

Configuring hardware monitoring for CPU, memory, and buffer usage

Alerts can be set for cpu, memory, and buffer usage.

When monitoring is configured, thresholds can be set. When the thresholds are exceeded, actions such as messages can be sent. Logs are saved for periods of time to enable viewing of threshold status.

NOTE
Support for the custom policy operand is not provided for CPU and memory threshold monitoring.

1. Enter global configuration mode.

```
device# configure terminal
```

2. To set the memory threshold between 40 and 60 and cause no message to be sent when thresholds are exceeded, enter the **threshold-monitor memory** command as follows.
3. To adjust cpu usage polling and retry attempts and cause a RASLog message to be sent when thresholds are exceeded, enter the **threshold-monitor cpu** command as follows.

Viewing threshold status

To view the status of currently configured thresholds, enter the **show running-config threshold-monitor** command, as follows:

NOTE
Default values are not displayed under the **show running-config threshold-monitor** command. Only custom values are displayed when a user applies a policy.

To display the default values of thresholds and alert options, enter the **show defaults threshold** command, as in the following example for interfaces.

```
device# show defaults threshold interface type Ethernet

Type: GigE-Port
+--------+-----------------------+----------------------+------+-------+
|        |    High Threshold     |    Low Threshold     |Buffer|Time   |
|Area    |Value | Above  | Below |Value | Above | Below |Value |Base   |
|        |      | Action | Action|      | Action| Action|      |       |
+--------+------+--------+-------+------+-------+-------+------+-------+
|MTC     |  300 | none   | none  |    12| none  | none  |     0|minute |
+--------+------+--------+-------+------+-------+-------+------+-------+
|CRCAlign|  300 | none   | none  |    12| none  | none  |     0|minute |
+--------+------+--------+-------+------+-------+-------+------+-------+
|Symbol  |    5 | none   | none  |     0| none  | none  |     0|minute |
+--------+------+--------+-------+------+-------+-------+------+-------+
|IFG     |  100 | none   | none  |     5| none  | none  |     0|minute |
+--------+------+--------+-------+------+-------+-------+------+-------+
MTC - Missing Termination Character
```

Cyclic redundancy check (CRC)

Cyclic redundancy check (CRC) polls CRC errors for each port in the configured polling interval.

If the number of CRC error exceeds the configured threshold in a polling window, the configured action is taken. You can set the threshold in the range 1 to 10.

NOTE
This feature is enabled by default. The default threshold is 5.

Port CRC supports following actions:

- **Raslog**: This is configured by default and the event are logged.
- **Port-shutdown**: If port-shutdown is configured as action, the event is logged and the port shuts down. The interface state changes to port CRC down. To bring up the port, you must explicitly enable the port.

The port CRC is enabled using the **crc enable** command. The command is run from the system monitor port configuration mode.

```
device (config-sys-mon-port)# crc ?
Possible completions:
  action          Set Port CRC Monitoring Action
  enable          Enable Port CRC Monitoring (Default: Enabled)
  poll-interval   Set Port CRC Monitoring Poll-Interval
  threshold       Set Port CRC Monitoring Threshold
```

The command **crc action** allows you to set various actions. The command **crc poll-interval** allows you to set the polling interval. The command **crc threshold** allows you to set the crc monitoring threshold.

The **show interface status** command displays the port crc status.

```
device# show interface status
--------------------------------------------------------------------------------
Port         Status          Mode       Speed    Type            Description
--------------------------------------------------------------------------------
Eth 0/1      connected (up)  --         10G      10G-SFP-SR
Eth 0/2      adminDown       --         --       --
Eth 0/3      notconnected    --         --       10G-SFP-SR
Eth 0/4      port-crcDown    --         --       --
```

To view port crc status on a specific ethernet interface, issue the **show interface ethernet** command.

```
device# show interface ethernet 0/4
Ethernet 0/4 is port-CRC down, line protocol is down (port-crc down)
Hardware is Ethernet, address is 00e0.0c76.79e8
    Current address is 00e0.0c76.79e8
Pluggable media not present
Interface index (ifindex) is 415367190
MTU 1548 bytes
10G Interface
LineSpeed Actual     : Nil
LineSpeed Configured : Auto, Duplex: Full
Priority Tag disable
Last clearing of show interface counters: 13:19:17
Queueing strategy: fifo
Receive Statistics:
    0 packets, 0 bytes
    Unicasts: 0, Multicasts: 0, Bro
```

You can also view the port crc status by issuing the **show ip interface brief** command.

```
device# show ip interface brief

Interface              IP-Address          Vrf                  Status                    Protocol
==================     ==========          ==================   ====================      ========
Port-channel 1         unassigned                               administratively down     down
Port-channel 2         unassigned                               administratively down     down
Ethernet 0/1           10.3.1.1            default-vrf          up                        up
Ethernet 0/2           unassigned          default-vrf          port-crc down             down
Ethernet 0/3           10.3.3.1            default-vrf          up                        down
Ethernet 0/4           unassigned          default-vrf          administratively down     down
```

To view the port crc status on a specific ip interface, issue the **ip interface ethernet** command.

```
device# show ip interface ethernet 0/4
 Ethernet 0/4 is port-crc down protocol is down
 IP unassigned
 Proxy Arp is not Enabled
 Vrf : default-vrf
```

Remote Monitoring

RMON overview

Remote monitoring (RMON) is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. The RMON specification defines a set of statistics and functions that can be exchanged between RMON-compliant console managers and network probes. As such, RMON provides you with comprehensive network-fault diagnosis, planning, and performance-tuning information.

Configuring and managing RMON

Both alarms and events are configurable RMON parameters.

- Alarms allow you to monitor a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms are paired with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- Events determine the action to take when an event is triggered by an alarm. The action can be to generate a log entry, an SNMP trap, or both. You must define the events before an alarm can be configured. If you do not configure the RMON event first, you will receive an error when you configure the alarm settings.

By default, no RMON alarms and events are configured and RMON collection statistics are not enabled.

Configuring RMON events

You can add or remove an event in the RMON event table that is associated with an RMON alarm number.

To configure RMON events, perform the following steps from privileged EXEC mode.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Configure SNMP v2c.

```
device(config)# snmp-server enable trap
device(config)# snmp-server view all 1 included
device(config)# snmp-server community public groupname group2
device(config)# snmp-server group group2 v2c read all write all notify all
sw0(config)# snmp-server user user2 groupname group2
sw0(config)# snmp-server host <host-name> public version 2c severity-level Info
sw0(config-host-<host-name>/public)# exit
```

3. Configure the RMON event for generating logs and traps.

```
device(config)# rmon event 100 description hi_100 log trap public owner john_smith
device(config)# rmon event 101 description lo_100 log trap public owner john_smith
```

4. Return to privileged EXEC mode.

```
device(config)# end
```

5. Save the *running-config* file to the *startup-config* file.

```
device# copy running-config startup-config
```

Configuring RMON Ethernet group statistics collection

You can collect RMON Ethernet group statistics on an interface. RMON alarms and events must be configured for you to display collection statistics. By default, RMON Ethernet group statistics are not enabled.

Ethernet group statistics collection is not supported on ISL links.

To collect RMON Ethernet group statistics on an interface, perform the following steps from privileged EXEC mode.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Enter the **interface** command to specify the interface type and slot/port number.

```
device(config)# interface ethernet 0/32:1
device(conf-if-eth-0/32:1)# no shutdown
```

3. Configure RMON collection history using the **rmon collection history** command.

```
device# configure terminal
device(config)# interface ethernet 0/1
device(conf-if-te-0/1)# rmon collection history 5 interval 120 owner admin
```

4. Configure RMON Ethernet group statistics on the interface.

```
device(conf-if-eth-0/32:1)# switchport
device(conf-if-eth-0/32:1)# rmon collection stats 100 owner john_smith
```

5. Return to privileged EXEC mode.

```
device(conf-if-eth-0/32:1)# end
```

6. Enter the **copy** command to save the running-config file to the startup-config file.

```
device# copy running-config startup-config
```

Configuring RMON alarm settings

To configure RMON alarms and events, perform the following steps from privileged EXEC mode.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Configure the RMON alarms.

   Following is an example of an alarm that tests every sample for rising threshold and falling threshold.

   ```
   device(config)# rmon alarm 100 1.3.6.1.2.1.16.1.1.1.5.100 interval 10 absolute rising-threshold
   10000 event 100 falling-threshold 5000 event 101 owner john_smith
   ```

   Delta alarms can be configured as follows.

   ```
   device(config)# rmon alarm 100 1.3.6.1.2.1.16.1.1.1.5.100 interval 5 delta rising-threshold 2000
   event 100 falling-threshold 1000 event 101 owner john_smith
   device(config)# rmon alarm 200 1.3.6.1.2.1.16.1.1.1.5.200 interval 5 delta rising-threshold 2000
   event 200 falling-threshold 1000 event 201 owner john_smith
   ```

   The events are not triggered when alarms are configured with invalid OIDS.

3. Return to privileged EXEC mode.

   ```
   device(config)# end
   ```

4. Save the *running-config* file to the *startup-config* file.

   ```
   device# copy running-config startup-config
   ```

5. To view configured alarms, use the **show running-config rmon alarm** command.

Monitoring CRC errors

Certain interface counters, such as those for CRC errors, may not be available by means of SNMP OIDs. In this case it is recommended that either RMON or CLI be used to monitor those statistics.

The following synchronizes the statistics maintained for the interface and RMON, as well as ensures proper reporting from an operational standpoint.

1. Issue the **clear counters all** command in global configuration mode.

   ```
   device# clear counters all
   ```

2. Issue the **clear counters rmon** command.

   ```
   device# clear counters rmon
   ```

3. Execute the **rmon collection stats** command on each interface, as in the following example.

   ```
   device(config)# interface ethernet 0/1
   device(conf-if-eth-0/1)# rmon collection stats 2 owner admin
   ```

   NOTE

   The CRC OID is 1.3.6.1.2.1.16.1.1.1.8. While configuring the alarm, this OID may be used for both delta and absolute alarms.

4. Use an appropriate RMON MIB for additional monitoring.
   For example, to obtain CRC statistics on a Extreme SLX-OS platform, the following RMON MIB could be used: Object-etherStatsCRCAlignErrors, OID- .1.3.6.1.2.1.16.1.1.1.8

Viewing RMON logs, events, alarms, and statistics

Following show commands enable you to view the RMON logs, events, alarms, and statistics.

- **show rmon logs**

```
device(config)# do show rmon logs
event Index = 100
        log Index = 1
        log generated time = 00:05:50 (35000)
        log description rising_100

event Index = 101
        log Index = 1
        log generated time = 00:05:30 (33000)
        log description falling_100
```

- **show rmon events**

```
device(config)# do show rmon events
event Index = 100
Description rising_100
        Event type Log
        Last Time Sent = 00:05:50
        Owner  john_smith

event Index = 101
Description falling_100
        Event type Log
        Last Time Sent = 00:05:30
        Owner  john_smith
```

- **show rmon alarms**

```
device(config)# do show rmon alarms
alarm Index = 100
        alarm status = VALID
        alarm Variable = 1.3.6.1.2.1.16.1.1.1.5.100
        alarm Interval = 10
        alarm Type is Absolute
        alarm Value  = 10000
        alarm Rising Threshold = 10000
        alarm Rising Event = 100
        alarm Falling Threshold = 5000
        alarm Falling Event = 101
        alarm Owner is john_smith
```

- **show rmon statistics**

```
device(config)# do show rmon statistics
rmon collection index 100
    Interface index is Id: 201589376 , Name : Ethernet 0/32:1
    Receive Statistics:
        10000 packets, 1280000 bytes,  0 packs dropped
        Multicasts: 0, Broadcasts: 0

        Under-size : 0, Jabbers: 0, CRC: 0
        Fragments: 0, Collisions: 0

        64 byte pkts: 0, 65-127 byte pkts: 0
        128-255 byte pkts: 10000, 256-511 byte pkts: 0
        512-1023 byte pkts: 0, 1024-1518 byte pkts: 0
        Over 1518-byte pkts(Oversize - Jumbo): 0

    Owner: RMON_SNMP
    Status: ok(1)
```

System Monitoring

System Monitor overview

System Monitor provides customizable monitoring thresholds, which allow you to monitor the health of each component of a device. Whenever a device component exceeds a configured threshold, System Monitor automatically provides notification by means of e-mail or RASLog messages, depending on the configuration.

Because of platform-specific values that vary from platform to platform, it was previously not possible to configure platform-specific thresholds through a global CLI command.

Threshold and notification configuration procedures are described in the following sections.

Monitored components

The following FRUs and temperature sensors are monitored on supported devices:

- **cid-card**—Displays the threshold for the chassis ID card component.
- **compact-flash**—Displays the threshold for the compact flash device.
- **fan**—Configures fan settings.
- **power**—Configures power supply settings.
- **temp**—Displays the threshold for the temperature sensor component.

NOTE
CID cards can be faulted and removed. The system continues to operate normally as long as one CID card is installed. If both CID cards are missing or faulted, the device will not operate.

Monitored FRUs

System Monitor monitors the absolute state of the following FRUs:

- Fan
- Power supply
- CID card

Possible states for all monitored FRUs are removed, inserted, on, off, and faulty. A state of none indicates the device is not configured. If the FRU is removed, inserted, or goes into a faulty state, System Monitor sends a RASLog message or an e-mail alert, depending on the configuration. The health status of the FRU being monitored is not affected by the on or off status. The System Monitor generates a separate RASLog message for the overall health of the device. Use the **show system monitor** command to view the health status of a device. Refer to the *Displaying the device health status* section for example output.

SFM monitoring

Switch Fabric Module (SFM), and Traffic Manager (TM) error interrupts are logged in RASLOG.

Configuring System Monitor

This section contains example basic configurations that illustrate various functions of the **system-monitor** command and related commands. For CLI details, refer to the *Command Reference* for your product.

Setting system thresholds

Each component can be in one of two states, down or marginal, based on factory-defined or user-configured thresholds. (The default thresholds are listed in Configuring System Monitor on page 40.)

1. Issue the **configure terminal** command to enter global configuration mode.
2. Change **down-threshold** and **marginal-threshold** values for the SFM.

> NOTE
> You can disable the monitoring of each component by setting **down-threshold** and **marginal-threshold** values to 0 (zero).

Setting state alerts and actions

System Monitor generates an alert when there is a change in the state from the default or defined threshold.

1. Issue the **configure terminal** command to enter global configuration mode.
2. To enable a RASLog alert (example: when the power supply is removed), enter the following command:

> NOTE
> There are no alerts for MM, compact-flash, or temp. There are no alert actions for SFPs.

Configuring e-mail alerts

Use the **system-monitor-mail fru** command to configure e-mail threshold alerts for FRU and optic monitoring. For an e-mail alert to function correctly, you must add the IP addresses and host names to the domain name server (DNS) in addition to configuring the domain name and name servers. For complete information on the **system-monitor-mail relay host** command, refer to the *Extreme SLX-OS Command Reference* for the SLX 9140 and 9240 Switches

1. Issue the **configure terminal** command to enter global configuration mode.
2. Enter the following command to enable e-mail alerts and to configure the e-mail address.

```
device(config)# system-monitor-mail fru enable email-id
```

Sendmail agent configuration

The sendmail agent must have one of the following configuration to resolve the domain-name.

- Configure DNS settings to connect device to DNS server.
- In case if DNS server is not available, DNS configuration along with relay host configuration is required for the sendmail agent on the device to resolve the domain-name. E-mail can be forwarded through the relay host. For example:

```
device(config)# ip dns domain-name corp.Extreme.com
device(config)# ip dns name-server 10.70.20.23
device(config)# ip dns name-server 10.70.20.43
```

The following **system-monitor-mail relay host** commands allow the sendmail agent on the device to resolve the domain name and forward all e-mail messages to a relay server.

- To create a mapping:

```
device(config)# system-monitor-mail relay 10.70.212.167 domain-name scooby.Extreme.com
```

- To delete the mapping:

```
device(config)# no system-monitor-mail relay 10.70.212.167 domain-name scooby.Extreme.com
```

- To change the domain name:

```
device(config)# system-monitor-mail relay 10.70.212.167 domain-name domain_name2.Extreme.com
```

  NOTE
  You must delete the first domain name before you can change it to a new domain name.

- To delete the domain name and return to the default:

```
device(config)# no system-monitor-mail relay 10.70.212.167 domain-name domain_name2.Extreme.com
```

Viewing system optical monitoring defaults

You can view the optical monitoring default values by entering **show defaults threshold** followed by the SFP type.

The following example command will display the defaults for type 1GLR SFPs:

```
device# show defaults threshold sfp type 1GLR
Type: 1GLR
+------------+------------------------+------------------------+--------+
|            |     High Threshold     |     Low Threshold      | Buffer |
| Area       | Value | Above  | Below | Value     |  Below     | Value  |
|            |       | Action | Action|           |  Action    |        |
+------------+-------+--------+-------+-----------+------------+--------+
| Temp C     |    90 | raslog | none  |        -45|  raslog    |       0|
+------------+-------+--------+-------+-----------+------------+--------+
| RXP uWatts |   501 | raslog | none  |          6|  raslog    |       0|
+------------+-------+--------+-------+-----------+------------+--------+
| TXP uWatts |   794 | raslog | none  |         71|  raslog    |       0|
+------------+-------+--------+-------+-----------+------------+--------+
| Current mA |    45 | raslog | none  |          1|  raslog    |       0|
+------------+-------+--------+-------+-----------+------------+--------+
| Voltage mV |  3700 | raslog | none  |       2900|  raslog    |       0|
+------------+-------+--------+-------+-----------+------------+--------+
device#
```

Viewing the area-wise optical monitoring current status

To view the area wise optical monitoring current status and value, run the **show threshold monitor sfp all area** command.

```
device# show threshold monitor sfp all area temperature
Interface                              Type            Area            Value           Status
Monitoring Status
-------------------------------------------------------------------------------------------------------------
-----------
Eth 0/5                                10GSR           Temperature     24 Centigrade   In Range
Monitoring
```

Displaying the device health status

To display the health status of a device, enter **show system monitor**.

```
device# show system monitor
** System Monitor Switch Health Report **
                        Switch status            : HEALTHY
                        Time of Report           : 2017-03-10 06:33:27
                        Power supplies monitor   : HEALTHY
                        Temperatures monitor     : HEALTHY
                        Fans monitor             : HEALTHY
                        Flash monitor            : HEALTHY
device#
```

Logging and tracing

Overview

Logging and tracing involves, RASTrace, RASLog, AuditLog and Syslog.

RASTrace captures low level info which can be used for debugging or troubleshooting issues. Use the **rasdecode** command to decode the traces collected. You must provide the module ID (-m) and display count (-n) parameters.

RASTrace, RASLog, AuditLog and Syslog are detailed in the following section of the docuemnt.

RASLog

RASLog subsystem provides centralized logging mechanism. RASLog messages log system events related to configuration changes or system error conditions.

It can store 2048 external customer visible messages in total. These are forwarded to the console, to the configured syslog servers and through the SNMP traps or informs the SNMP management station.

There are four levels of severity for messages, ranging from CRITICAL to INFO. In general, the definitions are wide ranging and are to be used as general guidelines for troubleshooting. You must look at each specific error message description thoroughly before taking action.

TABLE 3 Severity levels of the RASLog messages

Severity level	Description
CRITICAL	Critical-level messages indicate that the software has detected serious problems that cause a partial or complete failure of a subsystem if not corrected immediately; for example, a power supply failure or rise in temperature must receive immediate attention.
ERROR	Error-level messages represent an error condition that does not affect overall system functionality significantly. For example, error-level messages may indicate time-outs on certain operations, failures of certain operations after retries, invalid parameters, or failure to perform a requested operation.
WARNING	Warning-level messages highlight a current operating condition that must be checked or it may lead to a failure in the future. For example, a power supply failure in a redundant system relays a warning that the system is no longer operating in redundant mode unless the failed power supply is replaced or fixed.
INFO	Info-level messages report the current non-error status of the system components; for example, detecting online and offline status of an interface.

For more information on RASLog messages, refer the *Extreme SLX-OS Message Reference*.

AuditLog

AuditLog messages are classified into three types: DCM Configuration (DCMCFG), Firmware (FIRMWARE), and Security (SECURITY).

DCMCFG audits all the configuration changes in DB. FIRMWARE audit the events occurring during firmware download process. SECURITY audit any user-initiated security event for all management interfaces. Audit log messages are saved in the persistent storage. The storage has a limit of 1024 entries and will wrap around if the number of messages exceed the limit.

The SLX device can be configured to stream Audit messages to the specified syslog servers. Audit log messages are not forwarded to SNMP management stations.

Following are few sample outputs.

```
device(config)# sflow polling-interval 25
2016/06/02-08:48:39, [SFLO-1004], 1067, M1 | Active | DCE, INFO,
MMVM, Global sFlow polling interval is changed to 25.
2016/06/02-08:48:39, [SFLO-1006], 1068, M1 | Active | DCE, INFO,
MMVM, sFlow polling interval on port Ethernet 1/14 is changed to
25.

device# show logging auditlog reverse count 2
394 AUDIT,2016/06/02-08:48:39 (GMT), [DCM-1006], INFO, DCMCFG,
admin/admin/127.0.0.1/console/cli,, SLX9140-4, Event: database
commit transaction, Status: Succeeded, User command: "configure
config sflow polling-interval 25".
393 AUDIT,2016/06/02-08:40:57 (GMT), [SEC-3022], INFO, SECURITY,
root/root/172.22.224.196/telnet/CLI,, MMVM, Event: logout, Status:
success, Info: Successful logout by user [root].
```

For more information on AuditLog messages, refer to the *Extreme SLX-OS Message Reference* for the SLX 9140 and 9240 Switches.

Syslog

The syslog protocol allow devices to send event notification messages across IP networks to event message collectors, also known as syslog servers.

RASLog and AuditLog infrastructure makes use of Syslog service running on the SLX device to log messages into the local file system or to remote syslog server. All external RASLog messages and all Audit logs are sent to syslog server. SLX-OS uses **syslog-ng** which is an open source implementation of the syslog protocol for Unix and Unix-like systems. It runs over any of the following:

- UDP (default port 514)
- TLS (default port 6514)

A maximum of 4 syslog servers can be configured on any SLX device. These servers can have IPV4 or IPV6 address and reside in mgmt-vrf, default-vrf or user defined VRF. The **logging syslog-server** command enables the syslog event capturing on the syslog server. The IP address, VRF-name and port are the parameters used.

Following are sample syslog events captured at the syslog server.

```
There is no reference to MMVM in Davinci. It's been changed to "SLX".

Please see the example below
======================
Feb 15 18:10:11 10.24.71.121 SLX raslogd: [log@1588
value="RASLOG"][timestamp@1588 value="2017-02-15T18:10:11.087052"]
msgid@1588 value="NSM-1003"][seqnum@1588 value="1566"][attr@1588
value=" DCE | WWN 10:00:ffffffc4:ffffffff5:7c[severity@1588 value="INFO"]
[swname@1588 value="SLX"][arg0@1588 value="Ethernet 0/1" desc="InterfaceName"]
BOM Interface Ethernet 0/1 is link down.
Feb 15 18:10:11 10.24.71.121 SLX raslogd: [log@1588 value="AUDIT"]
[timestamp@1588 value="2017-02-15T18:10:11.112495"][tz@1588 value="GMT"]
```

```
[msgid@1588 value="DCM-1006"][severity@1588 value="INFO"]
[class@1588 value="DCMCFG"][user@1588 value="admin"][role@1588 value="admin"]
[ip@1588 value="127.0.0.1"][interface@1588 value="console"][application@1588 value="cli"]
[swname@1588 value="SLX9140"][arg0@1588 value="database commit transaction" desc="Event Name"]
[arg1@1588 value="Succeeded" desc="Command status"]
[arg2@1588 value=""configure conf-if-eth-0/1 shutdown"" desc="ConfD hpath string"]
BOMEvent: database commit transaction, Status: Succeeded, User command: "configure conf-if-eth-0/1
shutdown".
Feb 15 18:13:02 10.24.71.121 SLX raslogd: [log@1588 value="RASLOG"]
[timestamp@1588 value="2017-02-15T18:13:02.261680"][msgid@1588 value="NSM-1019"]
[seqnum@1588 value="1567"][attr@1588 value=" DCE | WWN 10:00:ffffffc4:fffffff5:7c[severity@1588
value="INFO"]
[swname@1588 value="SLX"][arg0@1588 value="Ethernet 0/1" desc="InterfaceName"]
BOM Interface Ethernet 0/1 is administratively up.
Feb 15 18:13:02 10.24.71.121 SLX raslogd: [log@1588 value="AUDIT"]
[timestamp@1588 value="2017-02-15T18:13:02.292919"]
[tz@1588 value="GMT"][msgid@1588 value="DCM-1006"]
[severity@1588 value="INFO"][class@1588 value="DCMCFG"]
[user@1588 value="admin"][role@1588 value="admin"][ip@1588 value="127.0.0.1"]
[interface@1588 value="console"][application@1588 value="cli"]
[swname@1588 value="SLX9140"][arg0@1588 value="database commit transaction" desc="Event Name"]
[arg1@1588 value="Succeeded" desc="Command status"]
[arg2@1588 value=""configure conf-if-eth-0/1 no shutdown"" desc="ConfD hpath string"]
BOMEvent: database commit transaction, Status: Succeeded, User command: "configure conf-if-eth-0/1 no
shutdown".
Feb 15 18:13:03 10.24.71.121 SLX raslogd: [log@1588 value="RASLOG"][timestamp@1588
value="2017-02-15T18:13:03.303235"]
[msgid@1588 value="NSM-1001"][seqnum@1588 value="1568"]
[attr@1588 value=" DCE | WWN 10:00:ffffffc4:fffffff5:7c[severity@1588 value="INFO"]
[swname@1588 value="SLX"][arg0@1588 value="Ethernet 0/1" desc="InterfaceName"] BOM Interface Ethernet 0/1
is online.
```

For more information on Syslog messages, refer to the *Extreme SLX-OS Message Reference* for the SLX 9140 and SLX 9240 Switches.

.

Importing a syslog CA certificate

The following procedure imports the syslog CA certificate from the remote host to the device.

1. Connect to the device and log in using an account with admin role permissions.
2. In privileged EXEC mode, enter the **certutil import syslogca** command. Include the full path to the certificate on the host, specify SCP as the protocol, and include the IP address of the host.

```
device# certutil import syslogca directory /usr/ldapcacert/ file cacert.pem protocol SCP host 10.23.24.56
user jane password
password: ****
```

Viewing the syslog CA certificate

The following procedure allows you to view the syslog CA certificate that has been imported on the device.

1. Connect to the device and log in using an account with admin role permissions.
2. In privileged EXEC mode, enter the **show cert-util syslogcacert** command.

This example displays the syslog CA certificates.

```
device# show cert-util syslogcacert
```

Verifying syslog CA certificates

To test whether a syslog CA certificate has been imported on the device, in privileged EXEC mode, enter the **no certutil syslogca** command and examine the message returned by the system. The command returns an error if there is no syslog CA certificate on the device. If a syslog CA certificate exists on the device, you are prompted to delete it. Enter the **no certutil syslogcacert** command to retain the certificate.

Example for when no syslog CA certificate is present

```
device# no certutil syslogcacert
% Error: syslog CA certificate does not exist.
```

Example for when a syslog CA certificate exists on the device

```
device# no certutil syslogcacert
Do you want to delete syslog CA certificate? [y/n]:n
```

Deleting a syslog CA certificate

The following procedure deletes the syslog CA certificates of all attached Active Directory servers from the device.

1. Connect to the device and log in using an account with admin role permissions.
2. In privileged EXEC mode, enter the **no certutil syslogca** command. You will be prompted to confirm that you want to delete the syslog CA certificates.

This example deletes the syslog CA certificates.

```
device# no certutil syslogca
Do you want to delete syslogca certificate? [y/n]:y
Warning: All the syslog CA certificates are deleted.
```

sFlow

sFlow overview

The sFlow protocol is an industry-standard technology for monitoring high-speed switched networks.

The sFlow standard consists of an sFlow agent that resides anywhere within the path of the packet and an sFlow collector that resides on a central server. This release is compliant with sFlow Version 5.

The sFlow agent combines the flow samples and interface counters into sFlow datagrams and forwards them to the sFlow Collector at regular intervals. The datagrams consist of information on, but not limited to, packet header, ingress interfaces, sampling parameters, and interface counters. Packet sampling is typically performed by the ASIC. The sFlow collector analyzes the sFlow datagrams received from different devices and produces a network-wide view of traffic flows. You can configure up to five collectors, using both IPv4 and IPv6 addresses.

The sFlow datagram provides information about the sFlow version, its originating agent's IP address, a sequence number, one or more flow samples or counter samples or both, and protocol information.

The sFlow agent uses two forms of operation:

- Time-based sampling of interface counters
- Statistical sampling of switched packets

sFlow can be port based or ACL based.

In port based sFlow, the sampling entity performs sampling on all flows originating from or destined to a specific port. Each packet is considered only once for sampling, irrespective of the number of ports it is forwarded to. Port based sFlow uses the port level sampling rate, if it is configured. Otherwise, it uses the global sampling rate. When port level sampling rate is unconfigured with 'no' option, it will revert back to using the global sampling rate.

Flow based sFlow ensures that sampling is done per flow instead of per port. Flow based sFlow uses sFlow profile sampling rate.

The following applications does flow based sFLow.

- User based sflow
- VxLAN visibility sflow

Port-based and flow-based sFlow are supported on physical ethernet ports only.

BGP AS-Path

The sFlow packet processing support for the sflow BGP AS path forwarding works when the BGP is up and it advertises routes. sFlow samples with destination IP (DIP) address and source IP (SIP) address that match the route in BGP routing table, collected and sent to the collector are appended with the BGP AS-path information also known as the extended gateway header. In case of samples with DIPs and SIPs that do not have route in BGP routing table and sent to sFlow Collector are not appended with AS-path. However, this does not impact the sFlow operation. This attribute identifies autonomous systems (ASs) through which update message has passed. The last AS traversed by prefix is placed at the beginning of list. You can configure a maximum number of 300 ASs.

NOTE

- By default, the BGP AS-path is enabled. It does not requires any specific configuration.
- After enbaling sFlow configuring sample collector, you must disable counter sampling globally or per interface.

BGP Community

A BGP community is used for traffic engineering and dynamic routing policies. It can be added to the route and advertised to all neighbors. The community attribute values are encoded with an Autonomous System (AS) number in the first two octets, with the remaining two octets defined by the AS. A prefix can have more than one community attribute. A BGP speaker that sees multiple community attributes in a prefix can act based on one, some or all the attributes. A router has the option to add or modify a community attribute before the router passes the attribute on to other peers. In sFlow, based on the standard, the community routing policies can be predicted for prefixes belonging to same community. An AS-path exists in an sFlow sample where a DIP matches the BGP route table but presence of community attribute is optional.

Feature support matrix for sFlow

The following table captures the sFlow feature support matrix for this release.

TABLE 4 sFlow feature support

sFlow Feature	Support
sFlow v5	Supported
sFlow MIB	Supported When the Data source related Table (sFlowFsTable) is retrieved, corresponding sFlowFsReceiver object will continue to return the first entry in the Collector table (sFlowRcvrTable).
Flow based sFlow	Supported Port-based and flow-based sFlow are supported on physical ethernet ports only.
Extended Gateway, Extended router, and NAT/MPLS/URL header formats	No Support for Extended Gateway. Only Raw header and Extended Switch header is supported.
sFlow scanning for inbound, outbound, or both directions on a port	Inbound only
Multiple collector configuration	A maximum of five IPv4 or IPv6 collectors could be configured and can be part of any of the configured VRFs.
Subagent-ID	Filled up with zero.
Agent IP address	Cannot be configured through CLI. Management IP is always used as the Agent IP address.
sFlow source interface	Supports configuration of source interface and corresponding IP of the interface will be the outer source IP of the sflow datagram reaching collector. This is not applicable for sflow packet reaching mgmt-vrf collector.
sFlow counter polling support on per-port, per-VLAN, or per-trunk or per tunnel basis	Supports per-port counter polling only.
Ability to disable sFlow counter polling	Supports global and per-interface level.
AS path cleanup timer (v4: BGP communities, v5: BGP next hop router)	Not supported

Configuring sFlow

sFlow configuration involves global configuration and configuration on interfaces. Following are the steps involved at a high level.

- Enable sFlow feature globally on the device.
- Configure sFlow collectors and optionally associated UDP ports.
- Configure ACL based sFlow or Enable sFlow forwarding on Physical interfaces.
- Configure other optional sFlow configuration parameters.

Configuring sFlow globally

Execute the following steps to configure sFlow globally.

1. Enter the configure terminal command to change to global configuration mode.

```
device# configure terminal
```

2. Enable the sFlow protocol globally.

```
device (config)# sflow enable
```

3. Configure sFlow collectors and optionally associated UDP ports.

```
device(config)# sflow collector 172.22.12.83 6343 use-vrf mgmt-vrf
device(config)# sflow collector fdd1:a123:b123:c123:34:1:1:2 4713 use-vrf vrf2
device(config)# sflow collector fdd1:a123:b123:c123:112:1:1:2 5566 use-vrf default-vrf
```

4. Set the sFlow polling interval (in seconds).

```
device(config)# sflow polling-interval 35
```

5. Set the sFlow sample-rate.

```
device(config)# sflow sample-rate 4096
```

6. (Optional) Configure sFlow source-interface.

```
device(config)# sflow source-interface ethernet 0/3
```

7. Return to privileged EXEC mode.

```
device(config)# end
```

8. Confirm the sFlow configuration status by using the show sflow or show sflow all commands.

```
device# show sflow
```

9. Clear any existing sFlow statistics to ensure accurate readings.

```
device# clear sflow statistics
```

10. (Optional) Issue the **show running-config sflow** command to view the configuration.

    Following is a sample **show running-config sflow** command output.

```
device# show running-config sflow
sflow enable
sflow source-interface ethernet 0/3
sflow collector 10.10.10.100 6343 use-vrf default-vrf
sflow collector 10.252.200.219 6343 use-vrf mgmt-vrf
sflow collector 172.22.224.199 6343 use-vrf mgmt-vrf
device# show running-config sflow-profile
sflow-profile profile1 sampling-rate 2
device#
```

Enabling flow-based sFlow

Perform the following steps, beginning in global configuration mode.

NOTE
The "deny ACL" rule is not supported for flow-based sflow. Only the permit action is supported.

1. Create an sFlow profile. Be sure to specify the sampling-rate as a power of 2.

```
device(config)# sflow-profile profile1 sampling-rate 256
```

2. Create a standard MAC ACL.

```
device# mac access-list standard acl1
device(conf-macl-std)# permit any
```

3. Create a class map and attach the ACL to the class map.

```
device(conf-macl-std)# class-map class1
device(config-classmap)# match access-group acl1
```

4. Create a policy map and attach the class map to the policy map.

```
device(config-classmap)# policy-map policy1
device(config-policymap)# class class1
```

5. Use the **sflow-profile** command to add an sFlow profile name.

   This example assigns the profile name "profile1."

```
device(config-policymap-class)# sflow-profile profile1
```

6. Switch to interface configuration mode.

```
device(config-policymap)# exit
device(config)# interface ethernet 0/1
device(conf-if-eth-0/1)#
```

7. Bind the policy map to an interface.

```
device(conf-if-eth-0/1)# service-policy in policy1
```

8. (Optional) Issue the **show running-config sflow-profile** command to view the configured sFlow profile.

```
device# show running-config sflow-profile
sflow-profile profile1 sampling-rate 256
device#
```

Disabling flow-based sFlow on specific interfaces

To disable sFlow on a specific interface, perform the following steps in interface configuration mode.

1. Switch to interface configuration mode.

```
device(config)# interface ethernet 0/1
device(conf-if-eth-0/1)#
```

2. Disable flow-based sFlow by removing the policy map.

```
device(conf-if-eth-0/1)# no service-policy in
```

3. Confirm the sFlow configuration status on the specific interface.

```
device# show sflow interface ethernet 0/1
```

Configuring sFlow for interfaces

After the global sFlow configuration, sFlow must be explicitly enabled on all the required interface ports.

NOTE
When sFlow is enabled on an interface port, it inherits the sampling rate and polling interval from the global sFlow configuration.

Enabling and customizing sFlow on specific interfaces

Perform the following steps in privileged EXEC mode to enable and customize sFlow on an interface. This task assumes that sFlow has already been enabled at the global level.

1. Enter the **interface** command.

```
device(config)# interface ethernet 0/1
```

2. Use the **sflow enable** command to enable sFlow on the interface.

```
device(conf-if-eth-0/1)# sflow enable
```

3. Configure the sFlow polling interval.

```
device(conf-if-eth-0/1)# sflow polling interval 35
```

4. Set the sFlow sample-rate.

```
device(conf-if-eth-0/1)# sflow sample-rate 8192
```

5. (Optional) Confirm the sFlow configuration status on the specified interface using the **show sFlow interface** command.

   Following is a sample output of the **show sFlow interface** command.

```
device# show sflow interface ethernet 0/1

sFlow info for interface Ethernet 0/1
-----------------------------------------------
Port based sflow services are:         enabled
Flow based sflow services are:         disabled
Configured sampling rate:              8192 pkts
Actual sampling rate:                  8192 pkts
Counter polling interval:              35 secs
Port backoffThreshold :                800
Sflow samples collected:               0
Counter samples collected :            14
device#
```

Configuring an sFlow policy map and binding it to an interface

Perform the following steps to configure an sFlow policy map and bind it to an interface.

1. Enter the **configure terminal** command to change to global configuration mode.

```
switch# configure terminal
```

2. Create a standard MAC access control list (ACL).

```
switch# mac access-list standard acl1
switch(conf-macl-std)# permit any
```

3. Create a class map and attach the ACL to the class map.

```
switch(conf-macl-std)# class-map class1
switch(config-classmap)# match access-group acl1
```

4. Create a policy map and attach the class map to the policy map.

```
switch(config-classmap)# policy-map policy1
switch(config-policymap)# class class1
```

5. Add an sFlow profile name by using the **sflow-profile**command.

   This example assigns the profile name "profile1"

```
switch(config-policymap-class)# sflow-profile policy1
```

6. Bind the policy map to an interface.

```
switch(conf-if-eth-0/1)# service-policy in policy1
```

Disabling sFlow on specific interfaces

NOTE

Disabling sFlow on the interface port does not completely shut down the network communication on the interface port.

To disable sFlow on a specific interface, perform the following steps in interface configuration mode.

1. Disable the sFlow interface.

```
switch(conf-if-eth-0/1)# no sflow enable
```

2. Return to privileged EXEC mode.

```
switch(conf-if-eth-0/1)# end
```

3. Confirm the sFlow configuration status on the specific interface.

```
switch# show sflow interface interface ethernet 0/1
```

sFlow agent address

An sFlow agent address provides the SNMP connectivity to the agent in accordance with the sFlowAgentAddress definition in the sFlow RFC. You can configure the sFlow agent address similar to the source IP configuration instead of using the default agent address, which is set to the IPv4 address of the management port of the active management module. Use the **sflow agent- address** command to configure the sFlow agent IPv4 or IPv6 address.

Configuring sFlow agent address

Use the following steps to configure the sFlow agent address.

> NOTE
> If the sFlow agent address is not configured, or if sFlow agent address configuration is removed or when the IP address of a specified interface corresponding to the specified agent-address IP address type is not configured, the IPv4 address of the management port of the active management module will be used.

From the global configuration mode, use the **sflow agent-address** to configure the sFlow agent address.

```
device# configure terminal
(device-config)# sflow agent-address ipv4 ethernet 2/5
```

Use the **no sflow agent-addres** command to removed the sFlow agent address configuration.

```
(device-config)# no sflow agent-address
```

Limitations and restrictions

The sFlow agent address feature has the following limitations.

- The **show sflow** command displays the IP address of the selected interface for agent-address field. It does not display the specified interface for agent-address field. Use the **show running-config** command to display the agent address interface information.

Configuration example

Global configuration

```
device(config)# sflow enable
2017/01/23-10:03:34, [SFLO-1001], 4337, DCE, INFO, switch, sFlow is enabled globally.
device(config)# no sflow enable
2017/01/23-10:03:53, [SFLO-1001], 4338, DCE, INFO, switch, sFlow is disabled globally.
device(config)# sflow sample-rate 4096
2017/01/23-10:04:15, [SFLO-1003], 4339, DCE, INFO, switch, Global sFlow sampling rate is changed to 4096.
device(config)# no sflow sample-rate
2017/01/23-10:04:45, [SFLO-1003], 4340, DCE, INFO, switch, Global sFlow sampling rate is changed to 2048.
device(config)# sflow polling-interval 30
2017/01/23-10:05:01, [SFLO-1004], 4341, DCE, INFO, switch, Global sFlow polling interval is changed to 30.
device(config)# no sflow polling-interval
2017/01/23-10:05:19, [SFLO-1004], 4342, DCE, INFO, switch, Global sFlow polling interval is changed to 20.
device(config)# sflow collector 172.22.108.57 6343
2017/01/23-10:06:00, [SFLO-1007], 4343, DCE, INFO, switch, 172.22.108.57 is configured as sFlow collector.
device(config)# sflow collector 10.1.15.2 6343 use-vrf default-vrf
2017/01/23-10:06:47, [SFLO-1007], 4344, DCE, INFO, switch, 10.1.15.2 is configured as sFlow collector.
device(config)# vrf red_vrf
device(config-vrf-red_vrf)# address-family ipv4 unicast
device(vrf-red_vrf-ipv4-unicast)# exit
device(config-vrf-red_vrf)# exit
device(config)# sflow collector 100.1.1.2 6343 use-vrf red_vrf
2017/01/23-10:08:42, [SFLO-1007], 4345, DCE, INFO, switch, 100.1.1.2 is configured as sFlow collector.
device(config)# do show sflow
sFlow services are:                     disabled
Global default sampling rate:           2048 pkts
Global default counter polling interval: 20 secs
Collector server address        Vrf-Name         Sflow datagrams sent
-------------------------------------------------------------------
10.1.15.2:6343                  default-vrf            0
100.1.1.2:6343                  red_vrf                0
172.22.108.57:6343              mgmt-vrf               0

device(config)# do show run sflow
sflow collector 10.1.15.2 6343 use-vrf default-vrf
sflow collector 100.1.1.2 6343 use-vrf red_vrf
sflow collector 172.22.108.57 6343 use-vrf mgmt-vrf
device(config)# no sflow collector 172.22.108.57
2017/01/23-10:12:38, [SFLO-1007], 4347, DCE, INFO, switch, 172.22.108.57 is unconfigured as sFlow collector.
device(config)# no sflow collector 10.1.15.2 6343 use-vrf default-vrf
2017/01/23-10:13:13, [SFLO-1007], 4348, DCE, INFO, switch, 10.1.15.2 is unconfigured as sFlow collector.
device(config)# no sflow collector 100.1.1.2 6343 use-vrf red_vrf
2017/01/23-10:13:54, [SFLO-1008], 4349, DCE, INFO, switch, All the sFlow collectors are unconfigured.
device(config)#
```

Interface configuration

```
device(config)# interface Ethernet 0/1
device(conf-if-eth-0/1)# sflow enable
2017/01/23-10:22:29, [SFLO-1002], 4351, DCE, INFO, switch, sFlow is enabled for port Ethernet 0/1.
device(conf-if-eth-0/1)# no sflow enable
2017/01/23-10:22:40, [SFLO-1002], 4352, DCE, INFO, switch, sFlow is disabled for port Ethernet 0/1.
device(conf-if-eth-0/1)# sflow enable
2017/01/23-10:23:13, [SFLO-1002], 4353, DCE, INFO, switch, sFlow is enabled for port Ethernet 0/1.
device(conf-if-eth-0/1)# sflow sample-rate 8192
2017/01/23-10:23:24, [SFLO-1005], 4354, DCE, INFO, switch, sFlow sampling rate on port Ethernet 0/1 is
changed to 8192.
device(conf-if-eth-0/1)# no sflow sample-rate
2017/01/23-10:23:34, [SFLO-1005], 4355, DCE, INFO, switch, sFlow sampling rate on port Ethernet 0/1 is
changed to 2048.
device(conf-if-eth-0/1)# sflow polling-interval 40
2017/01/23-10:25:01, [SFLO-1006], 4356, DCE, INFO, switch, sFlow polling interval on port Ethernet 0/1 is
changed to 40.
device(conf-if-eth-0/1)# no sflow polling-interval
2017/01/23-10:25:12, [SFLO-1006], 4357, DCE, INFO, switch, sFlow polling interval on port Ethernet 0/1 is
changed to 20.
device(conf-if-eth-0/1)#
```

Application Telemetry

Introduction to Application Telemetry

The primary purpose of Application Telemetry is to extract network analytics, for example, application name, flow pathing, bandwidth, and latency from Extreme Networks SLX switch platforms.

NOTE

This feature is supported only on the SLX 9140. You can use either sFlow or Application telemetry or both at the same time, as they can co-exist on a switch.

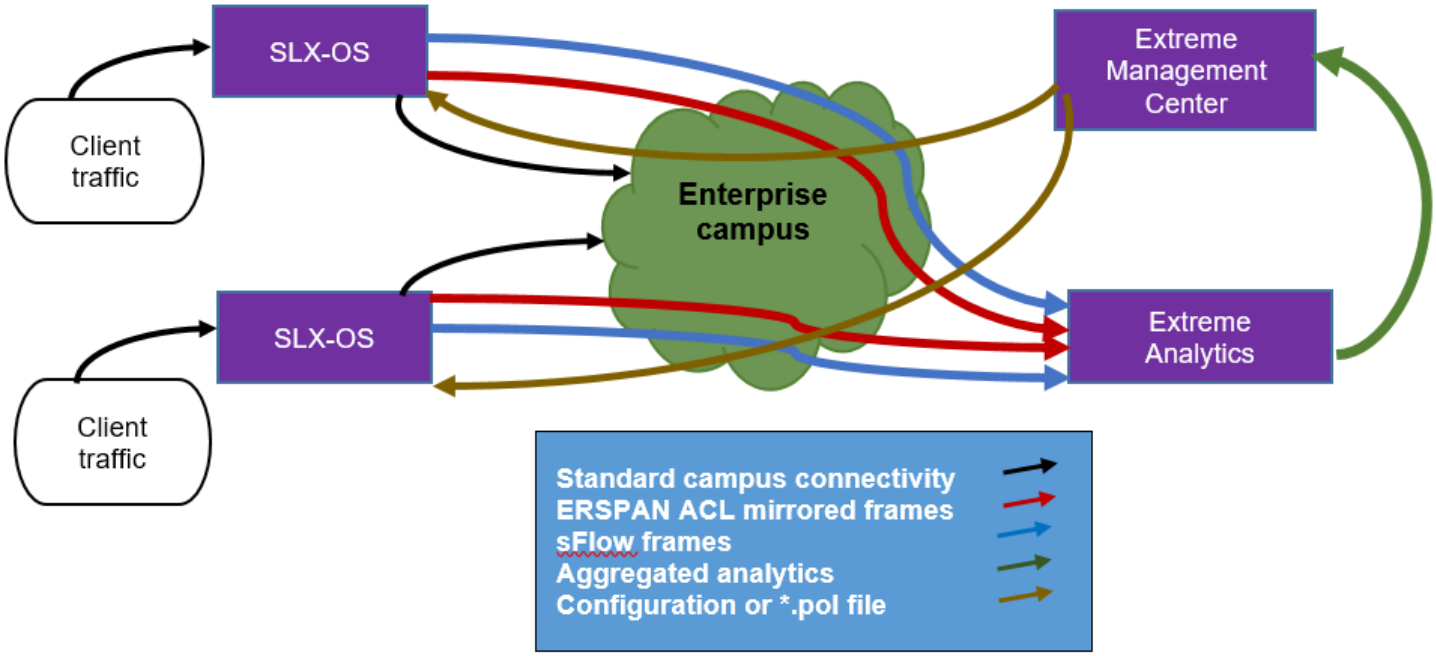
ATTENTION

sFlow must be enabled to use Application Telemetry.

Application Telemetry uses the sFlow and the Encapsulated Remote Switched Port Analyzer (ERSPAN) protocols to extract and transport specific raw packets and sampled flows from the SLX-OS switches to Extreme Analytics processing engines for further analysis. When a switch is added as a telemetry source, an Extreme Management Center (XMC) server runs a Tcl script that configures the switch automatically. Manual configuration is also supported.

The first set of raw traffic information is produced by highly specific ingress ACLs (processed within the SLX-OS hardware); these ACLs are applied at the system level (on all interfaces) of an SLX-OS switch to match specific packet types (for example, TCP SYN or DNS packets) with the purpose of mirroring these packets to the Extreme Analytics engine for further analysis. The ACL-filtered traffic is encapsulated and transported by means of the ERSPAN protocol towards Analytics Engines where, as with the ACLs, processing is handled by the SLX-OS hardware. The red connections in the logical diagram below represent the ERSPAN mirrored traffic between the SLX-OS switches and the Extreme Analytics appliance.

FIGURE 3 Application Telemetry overview



The second set of raw traffic is generated by the standard sFlow protocol and is enabled on all SLX-OS interfaces. By its very nature, sFlow is a sampled packet technology that is processed within the SLX-OS CPU. The sampled traffic is then transported over UDP to the analytics engine for further analysis. The sFlow feed is represented in the logical diagram above by the blue connections from the SLX-OS switches to the ExtremeAnalytics appliance.

The analytics engine processes the ERSPAN flows to extract application details, network flows, network response time for TCP-based flows, application response time for HTTP, HTTPS (SSL), DNS, DHCP, and so on. The sFlow information is used to deduce the bandwidth calculations of the individual flows and applications.

The result is that the application name, network response time, and application response time extracted from the ERSPAN mirrored traffic provides the basic Application Telemetry flow. When a sampled sFlow is matched to a basic flow, an enhanced Application Telemetry flow is produced that contains packet and byte counters, along with the details of a network to a switch interface.

Feature details

The following bullets summarize how the Application Telemetry feature works.

- The feature is enabled and disabled at the global level.
- A telemetry policy file is copied to the switch, by means of an XMC server script, over TFTP.
- SLX-OS saves 133 Application Telemetry filters in a telemetry.pol file.
- The IP address of the sFlow agent (the management IP address of the switch) and the IP address of the first collector in the default VRF acts as the source and destination for Generic Routing Encapsulation (GRE) configuration.
- Only one sFlow collector is supported for this feature. If multiple collectors are configured, the first collector configured with the default VRF is selected. This feature supports only the first sFlow IPv4 collector with the default VRF.
- If no sFlow collector is configured with the default VRF, an error is returned when the feature is enabled.

- The feature uses three new TCAM profiles, app-tele-l2-l3-iacl, app-tele-l3-iqos-l2-iacl and app-tele-l3-iqos-l3-iacl to optimize hardware resources. Refer **profile tcam** command for the new options.

Feature limitations

The following bullets summarize the limitations of the Application Telemetry feature:

- FlexACLs are used internally to support this feature and are not user configurable.
- ERSPAN encapsulation internally uses one hardware SPAN session out of four available sessions. If all hardware SPAN sessions are already exhausted and the user tries to enable this feature by means of **app-telemetry enable** command, an error message appears.
- When content of the app-telemetry.pol file is changed, user must remove and reapply file by using rules under the provided configuration command.
- When the switch is reloaded, the app-telemetry.pol file is read and the telemetry ACLs are installed if the configuration has been saved. If the correct telemetry profile is not loaded on the switch and the feature is enabled, an error message is issued.
- Telemetry rules and ACL statistics are not persistent following a system reload.

Enabling TCAM profiles for Application Telemetry

New TCAM profiles are provided to enable hardware resources for ingress application telemetry. These must be enabled as in the following examples.

The following example enables the profile to support IPv4, IPv6 QoS, and Layer 2 ACLs.

```
SLX(config-hardware)# profile tcam app-tele-l2-l3-iacl
```

The profile is as follows.

```
TCAM profile:    app-tele-l2-l3-iacl
_____________________________________________________________________
 MAC ACL Based QoS Policy Entries (Ingress):    0
         MAC Security ACL Entries (Ingress):    512
        MAC Policy Based forwarding entries:    0
                     L2 OpenFlow (Ingress):    0
IPV4 ACL Based QoS Policy Entries (Ingress):    0
IPV4 Policy Based Routing Entries (Ingress):    0
        IPV4 Security ACL Entries (Ingress):    0
                   IPV4 OpenFlow (Ingress):    0
IPV6 Policy Based Routing Entries (Ingress):    0
IPV6 ACL Based QoS Policy Entries (Ingress):    0
        IPV6 Security ACL Entries (Ingress):    0
          IP Security ACL Entries (Ingress):    1536
  IP ACL Based QoS Policy Entries (Ingress):    0
         MAC Security ACL Entries (Egress):    0
  MAC ACL Based QoS Policy Entries (Egress):    0
        IPV4 Security ACL Entries (Egress):    0
 IPV4 ACL Based QoS Policy Entries (Egress):    0
        IPV6 Security ACL Entries (Egress):    0
 IPV6 ACL Based QoS Policy Entries (Egress):    0
          IP Security ACL Entries (Egress):    0
   IP ACL Based QoS Policy Entries (Egress):    0
           Leaf Flex ACL Entries (Ingress):    1024
_____________________________________________________________________
```

The following example enables the profile to support IPv4, IPv6 QoS, and Layer 2 ACLs.

```
SLX(config-hardware)# profile tcam app-tele-l3-iqos-l2-iacl
```

The profile is as follows.

```
TCAM profile:    app-tele-l3-iqos-l2-iacl
_____________________________________________________
MAC ACL Based QoS Policy Entries (Ingress):     0
MAC Security ACL Entries (Ingress):           512
MAC Policy Based forwarding entries:            0
L2 OpenFlow (Ingress):                          0
IPV4 ACL Based QoS Policy Entries (Ingress):    0
IPV4 Policy Based Routing Entries (Ingress):    0
IPV4 Security ACL Entries (Ingress):            0
IPV4 OpenFlow (Ingress):                        0
IPV6 Policy Based Routing Entries (Ingress):    0
IPV6 ACL Based QoS Policy Entries (Ingress):    0
IPV6 Security ACL Entries (Ingress):            0
IP Security ACL Entries (Ingress):              0
IP ACL Based QoS Policy Entries (Ingress):      1536
MAC Security ACL Entries (Egress):              0
MAC ACL Based QoS Policy Entries (Egress):      0
IPV4 Security ACL Entries (Egress):             0
IPV4 ACL Based QoS Policy Entries (Egress):     0
IPV6 Security ACL Entries (Egress):             0
IPV6 ACL Based QoS Policy Entries (Egress):     0
IP Security ACL Entries (Egress):               0
IP ACL Based QoS Policy Entries (Egress):       0
Flex Policy Based Entries (Ingress):            1024
_____________________________________________________
```

The following example enables the profile to support IPv4, IPv6 QoS, and IPv4 and IPv6 ACLs.

```
SLX(config-hardware)# profile tcam app-tele-l3-iqos-l3-iacl
```

The profile is as follows.

```
TCAM profile:    app-tele-l3-iqos-l3-iacl
_____________________________________________________
MAC ACL Based QoS Policy Entries (Ingress):     0
MAC Security ACL Entries (Ingress):             0
MAC Policy Based forwarding entries:            0
L2 OpenFlow (Ingress):                          0
IPV4 ACL Based QoS Policy Entries (Ingress):    0
IPV4 Policy Based Routing Entries (Ingress):    0
IPV4 Security ACL Entries (Ingress):            0
IPV4 OpenFlow (Ingress):                        0
IPV6 Policy Based Routing Entries (Ingress):    0
IPV6 ACL Based QoS Policy Entries (Ingress):    0
IPV6 Security ACL Entries (Ingress):            0
IP Security ACL Entries (Ingress):              1536
IP ACL Based QoS Policy Entries (Ingress):      512
MAC Security ACL Entries (Egress):              0
MAC ACL Based QoS Policy Entries (Egress):      0
IPV4 Security ACL Entries (Egress):             0
IPV4 ACL Based QoS Policy Entries (Egress):     0
IPV6 Security ACL Entries (Egress):             0
IPV6 ACL Based QoS Policy Entries (Egress):     0
IP Security ACL Entries (Egress):               0
IP ACL Based QoS Policy Entries (Egress):       0
Flex Policy Based Entries (Ingress):            1024
_____________________________________________________
```

Application Telemetry configuration components

The feature requires both sFLow sampled flows and ERSPAN encapsulated flows. These can be configured by means of a script as well as manually.

The following components are required to enable the feature, and all are executed by means of a Tcl script at the XMC server. The user can also execute the CLI manually if needed (for example, to disable/enable the feature or reload the app-telemetry.pol file when it is updated with new policy rules).

Configuring sFlow

The following configuration sequence, executed by means of a script on the XMC server, enables sFlow on all ports, and configures the sFlow collector, sample rate, and polling interval.

```
SLX(config)# sflow enable
SLX(config)# sflow sample-rate 1024
SLX(config)# sflow collector 10.1.1.1 6343 use-vrf default-vrf
SLX(config)# sflow polling-interval 60

SLX(config)# int eth 0/1-54
SLX(conf-if-eth-0/1-54)# sflow enable
```

NOTE

For information about sFlow, refer to the "sFlow" chapter in the *Extreme SLX-OS Monitoring Configuration Guide*.

Obtaining a *.pol file from the XMC server

```
SLX# copy tftp://@<hostname>//<filepath> flash://<filename
```

This command copies the *.pol file to the following directory on the switch:

```
/var/config/vcs/scripts
```

Enabling the Application Telemetry feature

The following new command is used in the XMC server configuration script to enable the feature. It can also be enabled (and disabled, by means of the **no** keyword) manually.

```
SLX(config)# app-telemetry enable
```

This command triggers the following operations in the backend:

- The sFlow module provides ERSPAN encapsulation-related fields (SIP, DIP, SMAC, DMAC, VLAN, Egress VIF, VRF) to hardware to encapsulate specific flows with an IPv4 GRE header and transport the encapsulated packets towards the Analytics Engines. The sFlow collector and ERSPAN destination share same IPv4 address, so that the Analytics Engine receives both sFlow and ERSPAN frames. Unlike sFlow, the specific flow packets are ERSPAN encapsulated and transported by the hardware itself.
- The switch management IPv4 address is used as the SIP. The first configured collector's IPv4 address in the default VRF is used as DIP. The MAC address of the egress port of ERSPAN flow is used as the SMAC, and that of the next-hop router is used as the DMAC. ERSPAN encapsulated frames are IP routed towards the sFlow collector. Any next-hop-related changes are accounted for automatically.
- The ERSPAN Type I header is supported, and an additional 38 bytes are added to the original packets during ERSPAN IPv4 GRE encapsulation (14 bytes L2 header, 20 bytes IP header, 4 bytes GRE header). The "Don't fragment" bit in the IP header is

set to TRUE, to avoid any fragmentation of ERSPAN-encapsulated packets. Consequently, it is recommended that the user enable jumbo frames across the ERSPAN pathway.

Configuring Application Telemetry ACLs

The following new command is used in the XMC server configuration script to configure ACLs for the feature. This can also be configured manually.

```
SLX(config)# app-telemetry access-list app-telemetry.pol ingress
```

Policy rules are written in .pol file format. These rules are parsed and the respective ACLs are applied at the system level on the SLX switch to capture specific flows, which are forwarded by means of ERSPAN.

All entries in the file are configured in the FlexACL region, which can have up to 1024 entries.

The app-telemetry.pol file should already be present in the switch's flash memory. The policy file is read and parsed to extract the appropriate ACL rules that are to be installed in the hardware. The action for all ACLs is "span" with a reserved span id. Once all the above steps are executed, Extreme Analytics Engines start receiving both sFlow and ERSPAN frames and do further deep-packet inspection.

Displaying telemetry ACLs

The following new command can be used in an XMC server configuration script to display telemetry ACLs, as in the following example. It can also be executed manually.

```
SLX# show app-telemetry access-list

uda access-list dhcpv4 on system at Ingress
    seq 10 permit udp any range 67 68 any range 67 68 (Active)
uda access-list dnstcp on system at Ingress
    seq 10 permit tcp any eq 53 any (Active)
uda access-list tcpsynack on system at Ingress
    seq 10 permit tcp any any ack sync (Active)
uda access-list bjnp on system at Ingress
    seq 10 permit udp any any 0x424a4e50 0xffffffff 0x0 0x0 0x0 0x0 0x0 0x0 (Active)
uda access-list eset on system at Ingress
    seq 10 permit tcp any any 0x0 0x0 0xabcd0000 0xffff0000 0x0 0x0 0x0 0x0 (Active)
```

Displaying telemetry counters

The following new command can be used in an XMC server configuration script to display telemetry counters, as in the following example. It can also be executed manually.

```
SLX# show app-telemetry counter

================================================
        Application Telemetry Counters
================================================
Number of dhcpv4 packets: 1258, bytes: 72145
Number of tcpsyn packets: 457, bytes: 270000
```