



## Configuring sFlow

---

This document describes the sampled flow (sFlow) feature and configuration steps to implement sFlow.

- [Prerequisites for Configuring sFlow, page 1](#)
- [Restrictions for Configuring sFlow, page 1](#)
- [Information About sFlow, page 1](#)
- [How to Provision sFlow, page 2](#)
- [Verifying sFlow, page 11](#)
- [Additional References, page 13](#)

### Prerequisites for Configuring sFlow

- You must enable sFlow on an interface on ME 1200 NID.
- NID must have an IP address.

### Restrictions for Configuring sFlow

- ME 1200 NID does not support configuring more than one sFlow instance with maximum sample rate on the specified data source, either in the ingress or egress direction.

### Information About sFlow

Using sFlow, a standards-based protocol mechanism, allows you to monitor Layer 2 traffic in data networks that contain switches and routers. It consists of :

- **sFlow Agent** (embedded on ME 1200 NID)—The sFlow Agent uses sampling technology to capture traffic statistics from the monitored device and then forwards the sampled data to a central sFlow Collector for analysis. Packet sampling is done using one or more sFlow instances, each configured with a sampling rate.

- sFlow Instances—There may be one or more sFlow Instances associated with a single data source. Each sFlow instance operates independently of other sFlow instances. For example, Packet Flow Sampling instances have their own sampling rates and Counter Sampling instances have their own sampling intervals.
- **sFlow Collector**—The sFlow Collector is a software application that can receive sFlow datagrams and present a view of traffic and other network parameters which are output as type, length, and value (TLV) in the datagrams. The sFlow collectors can also read and configure sFlow-managed objects. Both counter and packet flow statistics are collected and sent as sFlow Datagrams (defined by maximum datagram size of 200-1468) to a sFlow Collector.
  - sFlow Datagram—The sFlow Datagram format specifies a standard format for the sFlow Agent to send sampled data to a remote sFlow Collector. The sFlow Datagram version 5 is supported. The format of the sFlow Datagram is specified using the External Data Representation (XDR) standard. This makes it simpler for the sFlow Agent to encode and the sFlow Collector to decode. Samples are sent as User Datagram Protocol (UDP) packets to the host and port specified in the SFLOW MIB or CLI. The assigned port for sFlow (and the default specified in the SFLOW MIB) is port 6343. All sFlow Agents and applications by default must use UDP port 6343.

By default, sFlow is disabled on ME 1200 NID. You can enable sFlow on a specific interface or port.

## How to Provision sFlow

### Enabling sFlow Globally

#### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>ConfigureNID</b>  <b>Example:</b> UCS# Configure NID 1	Opens a new session for NID 1.
<b>Step 2</b>	<b>sflow</b>  <b>Example:</b> UCS# sflow	Enters the sFlow mode.
<b>Step 3</b>	<b>Flow_global {agent-ip {ipv4   ipv6}   collector-ip {ipv4   ipv6}   collector-port   datagram-maxsize   rx-timeout}</b>  <b>Example:</b> UCS(SFlow)# setsFlowGlobalConfig sFlow_global agent-ip ipv4 7.25.16.63 UCS(SFlow)# setsFlowGlobalConfig sFlow_global collector-ip ipv4 7.25.16.253	Enters the sFlow global configuration mode. <ul style="list-style-type: none"> <li>• <b>agent-ip</b>—Specifies Agent IP address.               <ul style="list-style-type: none"> <li>◦ <b>ipv4</b>—Specifies IPv4 address.</li> <li>◦ <b>ipv6</b>—Specifies IPv6 address.</li> </ul> </li> <li>• <b>collector-ip</b>—Specifies collector IP address.</li> </ul>

	Command or Action	Purpose
	<pre>UCS(SFlow)# setsFlowGlobalConfig sFlow_global collector-port 2033 UCS(SFlow)# setsFlowGlobalConfig sFlow_global datagram-maxsize 512 UCS(SFlow)#setsFlowGlobalConfig sFlow_global rx-timeout 50000</pre>	<ul style="list-style-type: none"> <li>◦ <b>ipv4</b>—Specifies IPv4 address.</li> <li>◦ <b>ipv6</b>—Specifies IPv6 address.</li> <li>• <b>collector-port</b>—Specifies collector UDP port. The valid range is from 1 to 65535.</li> <li>• <b>datagram-maxsize</b>—Specifies maximum datagram size. The valid range is from 200 to 1468.</li> <li>• <b>rx-timeout</b>—Specifies the receive timeout in seconds. The valid range is from 0 to 2147483647. The switch decrements the timeout once every second, and samples are received as long as it is non-zero. Once it reaches zero, receiver and all its configurations are reset to defaults.</li> </ul>
<b>Step 4</b>	<p><b>setsFlowGlobalConfig review</b></p> <p><b>Example:</b> UCS(SFlow)# setsFlowGlobalConfig review</p>	(Optional) Displays the configuration.
<b>Step 5</b>	<p><b>setsFlowGlobalConfig commit</b></p> <p><b>Example:</b> UCS(SFlow)# setsFlowGlobalConfig commit</p>	Sends the configuration to NID.
<b>Step 6</b>	<p><b>exit</b></p> <p><b>Example:</b> UCS(SFlow)# exit</p>	Exits the SFlow mode.

### Configuration Example

- The example shows how to enable sFlow globally:

```
UCS# sflow
UCS(SFlow)# setsFlowGlobalConfig sFlow_global agent-ip ipv4 7.25.16.63
UCS(SFlow)# setsFlowGlobalConfig sFlow_global collector-ip ipv4 7.25.16.253
UCS(SFlow)# setsFlowGlobalConfig sFlow_global collector-port 2033
UCS(SFlow)# setsFlowGlobalConfig sFlow_global datagram-maxsize 512
UCS(SFlow)# setsFlowGlobalConfig sFlow_global rx-timeout 50000
UCS(SFlow)# setsFlowGlobalConfig review
```

Commands in queue:

```
setsFlowGlobalConfig sFlow_global agent-ip ipv4 7.25.16.63
setsFlowGlobalConfig sFlow_global collector-ip ipv4 7.25.16.253

setsFlowGlobalConfig sFlow_global collector-port 6343
setsFlowGlobalConfig sFlow_global datagram-maxsize 512
setsFlowGlobalConfig sFlow_global rx-timeout 50000
```

```
UCS(SFlow)# setsFlowGlobalConfig commit
```

```
SetsFlowGlobalConfig Commit Success!!!
UCS(SFlow)# exit
```

## Enabling sFlow on a Port

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>ConfigureNID</b>  <b>Example:</b> UCS# Configure NID 1	Opens a new session for NID 1.
Step 2	<b>sflow</b>  <b>Example:</b> UCS# sflow	Enters the sFlow mode.
Step 3	<b>sflow_port {interface-id   enable   flow-sampler {enable-defaults   sampling-rate   sampling-maxsize}   counter-poller {enable   interval}}</b>  <b>Example:</b> UCS(SFlow)# setsFlowPortConfig sFlow_port interface-id 3 UCS(SFlow)# setsFlowPortConfig sFlow_port enable enable UCS(SFlow)# setsFlowPortConfig sFlow_port counter-poller enable enable UCS(SFlow)# setsFlowPortConfig sFlow_port counter-poller interval 10 UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler enable-defaults enable	Enters the sFlow port specific configuration mode. <ul style="list-style-type: none"> <li>• <b>interface-id</b>—Specifies physical port.</li> <li>• <b>enable</b>—Enables or disables sFlow on this port.</li> <li>• <b>flow-sampler</b>—Specifies sFlow flow sampler configuration.               <ul style="list-style-type: none"> <li>◦ <b>enable-defaults</b>—Enables the flow sampler default values.</li> <li><b>Note</b> To configure <b>sampling-rate</b> and <b>sampling-maxsize</b> as per your requirement, you must set this option to disable.</li> <li>◦ <b>sampling-rate</b>—Specifies the statistical sampling rate. The valid range is from 1 to 4294967295.</li> <li>◦ <b>sampling-maxsize</b>—Specifies maximum number of bytes to transmit per flow sample. The valid range is from 14 to 200.</li> </ul> </li> <li>• <b>counter-poller</b>—Specifies Interface counter poller configuration.               <ul style="list-style-type: none"> <li>◦ <b>enable</b>—Enables counter poller.</li> <li>◦ <b>interval</b>—Specifies counter poll interval. The valid range is from 1 to 3600 seconds.</li> </ul> </li> </ul>

	Command or Action	Purpose
<b>Step 4</b>	<b>setsFlowPortConfig review</b>  <b>Example:</b> UCS(SFlow)# setsFlowPortConfig review	(Optional) Displays the configuration.
<b>Step 5</b>	<b>setsFlowPortConfig commit</b>  <b>Example:</b> UCS(SFlow)# setsFlowPortConfig commit	Sends the configuration to NID.
<b>Step 6</b>	<b>exit</b>  <b>Example:</b> UCS(SFlow)# exit	Exits the SFlow mode.

### Configuration Example



#### Note

sFlow configuration does not persist on the NID. Running **show running-config.xml** command does not display the sFlow configuration globally or per-port. This is working as designed.

- The example shows how to enable sFlow on a port with default values enabled:

```
UCS# sflow
UCS(SFlow)# setsFlowPortConfig sFlow_port interface-id 3
UCS(SFlow)# setsFlowPortConfig sFlow_port enable enable
UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler enable-defaults enable
UCS(SFlow)# setsFlowPortConfig review
```

Commands in queue:

```
setsFlowPortConfig sFlow_port interface-id 3
setsFlowPortConfig sFlow_port enable enable
setsFlowPortConfig sFlow_port flow-sampler enable-defaults
enable
```

```
UCS(SFlow)# setsFlowPortConfig commit
```

```
SetsFlowPortConfig Commit Success!!!
```

```
UCS(SFlow)# exit
```

- The example shows how to enable sFlow on a port without any default values set:

```
UCS# sflow
UCS(SFlow)# setsFlowPortConfig sFlow_port interface-id 1
UCS(SFlow)# setsFlowPortConfig sFlow_port enable enable
UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler enable-defaults disable
UCS(SFlow)# setsFlowPortConfig review
```

Commands in queue:

```
setsFlowPortConfig sFlow_port interface-id 1
setsFlowPortConfig sFlow_port enable enable
```

```
setsFlowPortConfig sFlow_port flow-sampler enable-defaults
disable
```

```
UCS(SFlow)# setsFlowPortConfig commit
```

```
SetsFlowPortConfig Commit Success!!!
```

```
UCS(SFlow)# exit
```

- The example shows how enable sFlow on a port with user-configured parameters:

```
UCS# sflow
UCS(SFlow)# setsFlowPortConfig sFlow_port interface-id 1
UCS(SFlow)# setsFlowPortConfig sFlow_port enable enable
UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler enable-defaults disable
UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler sampling-maxsize 512
UCS(SFlow)# setsFlowPortConfig sFlow_port flow-sampler sampling-rate 200
UCS(SFlow)# setsFlowPortConfig sFlow_port counter-poller enable enable
UCS(SFlow)# setsFlowPortConfig sFlow_port counter-poller interval 30
UCS(SFlow)# setsFlowPortConfig review
```

Commands in queue:

```
setsFlowPortConfig sFlow_port interface-id 1
setsFlowPortConfig sFlow_port enable enable
setsFlowPortConfig sFlow_port flow-sampler enable-defaults
disable
setsFlowPortConfig sFlow_port flow-sampler sampling-maxsize
512
setsFlowPortConfig sFlow_port flow-sampler sampling-rate 200

setsFlowPortConfig sFlow_port counter-poller enable enable
setsFlowPortConfig sFlow_port counter-poller interval 30
```

```
UCS(SFlow)# setsFlowPortConfig commit
```

```
SetsFlowPortConfig Commit Success!!!
```

```
UCS(SFlow)# exit
```

## Getting Current Global sFlow Values

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>ConfigureNID</b>  <b>Example:</b> UCS# Configure NID 1	Opens a new session for NID 1.
<b>Step 2</b>	<b>sflow</b>  <b>Example:</b> UCS# sflow	Enters the sFlow mode.

	Command or Action	Purpose
<b>Step 3</b>	sFlow_global_req  <b>Example:</b> UCS(SFlow)# getsFlowGlobalConfig sFlow_global_req	Enters the sFlow global configuration mode.
<b>Step 4</b>	<b>getsFlowGlobalConfig review</b>  <b>Example:</b> UCS(SFlow)# getsFlowGlobalConfig review	(Optional) Displays the configuration.
<b>Step 5</b>	<b>getsFlowGlobalConfig commit</b>  <b>Example:</b> UCS(SFlow)# getsFlowGlobalConfig commit	Sends the configuration to NID.
<b>Step 6</b>	<b>exit</b>  <b>Example:</b> UCS(SFlow)# exit	Exits the SFlow mode.

### Configuration Example

- The example shows how to get current global sFlow values:

```
UCS# sflow
UCS(SFlow)# getsFlowGlobalConfig sFlow_global_req
UCS(SFlow)# getsFlowGlobalConfig review
```

Commands in queue:

```
getsFlowGlobalConfig sFlow_global_req
```

```
UCS(SFlow)# getsFlowGlobalConfig commit
```

```
GetsFlowGlobalConfig_Output.sFlow_global.agent_ip.t = 1
GetsFlowGlobalConfig_Output.sFlow_global.agent_ip.u.ipv4 = '0.0.0.0'
GetsFlowGlobalConfig_Output.sFlow_global.collector_ip.t = 1
GetsFlowGlobalConfig_Output.sFlow_global.collector_ip.u.ipv4 =
'0.0.0.0'
GetsFlowGlobalConfig_Output.sFlow_global.collector_port = 65535
GetsFlowGlobalConfig_Output.sFlow_global.datagram-maxsize = 1468
GetsFlowGlobalConfig_Output.sFlow_global.rx-timeout = 50000
```

```
GetsFlowGlobalConfig Commit Success!!!
```

```
UCS(SFlow)# exit
```

The following is a sample output on the NID.

```
Decoding of Request message was successful urn:#getsFlowConfig
Decoded record:
GetsFlowGlobalConfig_Input.sFlow_global_req = '0'
Encoding of Response message was successful
Encoded record:
```

```

GetsFlowGlobalConfig_Output.sFlow_global.agent_ip.t = 1
GetsFlowGlobalConfig_Output.sFlow_global.agent_ip.u.ipv4 = '0.0.0.0'
GetsFlowGlobalConfig_Output.sFlow_global.collector_ip.t = 1
GetsFlowGlobalConfig_Output.sFlow_global.collector_ip.u.ipv4 =
'0.0.0.0'
GetsFlowGlobalConfig_Output.sFlow_global.collector_port = 65535
GetsFlowGlobalConfig_Output.sFlow_global.datagram-maxsize = 1468
GetsFlowGlobalConfig_Output.sFlow_global.rx-timeout = 50000
GetsFlowGlobalConfig_Output.xmlns:ns0 =
"http://new.webservice.namespace"
GetsFlowGlobalConfig_Output.xmlns:http =
"http://schemas.xmlsoap.org/wsdl/http/"
GetsFlowGlobalConfig_Output.xmlns:mime =
"http://schemas.xmlsoap.org/wsdl/mime/"
GetsFlowGlobalConfig_Output.xmlns:soap =
"http://schemas.xmlsoap.org/wsdl/soap/"
GetsFlowGlobalConfig_Output.xmlns:soapenc =
"http://schemas.xmlsoap.org/soap/encoding/"
GetsFlowGlobalConfig_Output.xmlns:wsdl =
"http://schemas.xmlsoap.org/wsdl/"

```

## Getting Current Port Specific sFlow Values

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>ConfigureNID</b>  <b>Example:</b> UCS# Configure NID 1	Opens a new session for NID 1.
<b>Step 2</b>	<b>sflow</b>  <b>Example:</b> UCS# sflow	Enters the sFlow mode.
<b>Step 3</b>	<b>sflowPortConfigReq <i>port id</i></b>  <b>Example:</b> UCS(SFlow)# getsFlowPortConfig sflowPortConfigReq 2	Enters the sFlow port-specific configuration mode.
<b>Step 4</b>	<b>getsFlowGlobalConfig review</b>  <b>Example:</b> UCS(SFlow)# getsFlowGlobalConfig review	(Optional) Displays the configuration.
<b>Step 5</b>	<b>getsFlowGlobalConfig commit</b>  <b>Example:</b> UCS(SFlow)# getsFlowGlobalConfig commit	Sends the configuration to NID.



	Command or Action	Purpose
Step 6	<b>exit</b>  <b>Example:</b> UCS(SFlow)# exit	Exits the SFlow mode.

### Configuration Example

- The example shows how to get current port-specific sFlow values:

```
UCS# sflow
UCS(SFlow)# getFlowPortConfig sflowPortConfigReq 2
UCS(SFlow)# getFlowGlobalConfig review
```

Commands in queue:

```
getFlowPortConfig sFlowPortConfigReq 2
```

```
UCS(SFlow)# getFlowGlobalConfig commit
```

```
GetFlowPortConfig_Output.sFlow_port.interface_id = 2
GetFlowPortConfig_Output.sFlow_port.enable = false
GetFlowPortConfig_Output.sFlow_port.flow_sampler.enable_defaults =
true
GetFlowPortConfig_Output.sFlow_port.flow_sampler.sampling_rate = 4096
GetFlowPortConfig_Output.sFlow_port.flow_sampler.sampling-maxsize =
128
GetFlowPortConfig_Output.sFlow_port.counter_poller.enable = false
GetFlowPortConfig_Output.sFlow_port.counter_poller.interval = 60

GetFlowPortConfig Commit Success!!!
```

```
UCS(SFlow)# exit
```

The following is a sample output on the NID.

```
GetFlowPortConfig_Input.sFlowPortConfigReq = 2
Encoding of Response message was successful
Encoded record:
GetFlowPortConfig_Output.sFlow_port.interface_id = 2
GetFlowPortConfig_Output.sFlow_port.enable = false
GetFlowPortConfig_Output.sFlow_port.flow_sampler.enable_defaults =
true
GetFlowPortConfig_Output.sFlow_port.flow_sampler.sampling_rate = 4096
GetFlowPortConfig_Output.sFlow_port.flow_sampler.sampling-maxsize =
128
GetFlowPortConfig_Output.sFlow_port.counter_poller.enable = false
GetFlowPortConfig_Output.sFlow_port.counter_poller.interval = 60
GetFlowPortConfig_Output.xmlns:ns0 = "http://new.webservice.namespace"
GetFlowPortConfig_Output.xmlns:http =
"http://schemas.xmlsoap.org/wsdl/http/"
GetFlowPortConfig_Output.xmlns:mime =
"http://schemas.xmlsoap.org/wsdl/mime/"
GetFlowPortConfig_Output.xmlns:soap =
"http://schemas.xmlsoap.org/wsdl/soap/"
```

```

GetsFlowPortConfig_Output.xmlns:soapenc =
"http://schemas.xmlsoap.org/soap/encoding/"
GetsFlowPortConfig_Output.xmlns:wsi =
"http://schemas.xmlsoap.org/wsi/"

```

## Clearing sFlow Statistics

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>ConfigureNID</b>  <b>Example:</b> UCS# Configure NID 1	Opens a new session for NID 1.
<b>Step 2</b>	<b>sflow</b>  <b>Example:</b> UCS# sflow	Enters the sFlow mode.
<b>Step 3</b>	<b>clear_sflow_stats</b>  <b>Example:</b> UCS(SFlow)# clearsFlowStatistics clear_sflow_stats	Clears sFlow statistics.
<b>Step 4</b>	<b>clearsFlowStatistics review</b>  <b>Example:</b> UCS(SFlow)# clearsFlowStatistics review	(Optional) Displays the configuration.
<b>Step 5</b>	<b>clearsFlowStatistics commit</b>  <b>Example:</b> UCS(SFlow)# clearsFlowStatistics commit	Sends the configuration to NID.
<b>Step 6</b>	<b>exit</b>  <b>Example:</b> UCS(SFlow)# exit	Exits the SFlow mode.

### Configuration Example

- The example shows how to clear sFlow statistics:

```

UCS# sflow
UCS((SFlow)# clearsFlowStatistics clear_sflow_stats
UCS((SFlow)# clearsFlowStatistics review

```

```

Commands in queue:
clearsFlowStatistics clear_sflow_stats

```

```
UCS(SFlow)# clearsFlowStatistics commit

ClearsFlowStatistics Commit Success!!!

UCS(SFlow)# exit
```

## Verifying sFlow

Use the following commands to verify the sFlow status on the UCS controller.

- `showsFlowStatistics sFlow_stats_req`

This command displays the sFlow statistics on the NID. The following is a sample output from the command:

```
UCS(SFlow)# showsFlowStatistics sFlow_stats_req
```

```
UCS(SFlow)# showsFlowStatistics review
```

```
Commands in queue:
    showsFlowStatistics sFlow_stats_req
```

```
UCS(SFlow)# showsFlowStatistics commit
```

```
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].interface_id
= 1
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].counter_samples
= 42
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].interface_id
= 2
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].interface_id
= 3
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].tx_flow_samples
= 1
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].interface_id
= 4
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].interface_id
= 5
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].interface_id
= 6
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].tx_flow_samples
= 0
```

```
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].tx_successes
= 0
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].tx_errors
= 43
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].flow_samples
= 1
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].counter_samples
= 42

ShowsFlowStatistics Commit Success!!!
```

The following is a sample output on the NID.

```
Decoding of Request message was successful urn:#showsFlowStatistics
Decoded record:
ShowsFlowStatistics_Input.sFlow_stats_req = '0'
Encoding of Response message was successful
Encoded record:
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].interface_id
= 1
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[0].counter_samples
= 42
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].interface_id
= 2
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[1].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].interface_id
= 3
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].tx_flow_samples
= 1
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[2].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].interface_id
= 4
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[3].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].interface_id
= 5
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[4].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].interface_id
= 6
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].tx_flow_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.samplers.interface_[5].counter_samples
= 0
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].tx_successes
= 0
```

```
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].tx_errors
= 43
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].flow_samples
= 1
ShowsFlowStatistics_Output.sFlow_stats.receiver.statistics[0].counter_samples
= 42
ShowsFlowStatistics_Output.xmlns:ns0 =
"http://new.webservice.namespace"
ShowsFlowStatistics_Output.xmlns:http =
"http://schemas.xmlsoap.org/wsdl/http/"
ShowsFlowStatistics_Output.xmlns:mime =
"http://schemas.xmlsoap.org/wsdl/mime/"
ShowsFlowStatistics_Output.xmlns:soap =
"http://schemas.xmlsoap.org/wsdl/soap/"
ShowsFlowStatistics_Output.xmlns:soapenc =
"http://schemas.xmlsoap.org/soap/encoding/"
ShowsFlowStatistics_Output.xmlns:wSDL =
"http://schemas.xmlsoap.org/wsdl/"
```

## Additional References

### Related Documents

Related Topic	Document Title
Cisco ME 3800x and ME 3600x Switches Software Configuration Guide, Cisco IOS Release 15.4(1)S	<a href="http://www.cisco.com/c/en/us/td/docs/switches/metro/me3600x_3800x/software/release/15-4_1_S/configuration/guide/3800x3600xscg.html">http://www.cisco.com/c/en/us/td/docs/switches/metro/me3600x_3800x/software/release/15-4_1_S/configuration/guide/3800x3600xscg.html</a>

### MIBs

MIB	MIBs Link
MIBs Supporting Cisco IOS	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a>

**Technical Assistance**

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p><a href="http://www.cisco.com/support">http://www.cisco.com/support</a></p>