Implementing Video Monitoring Service on Cisco ASR 9000 Series Routers

This module describes how to implement video monitoring on Cisco ASR 9000 Series Aggregation Services Routers running Cisco IOS XR software.

Video monitoring is a service to monitor application (mainly video) traffic quality by measuring per-flow statistics on the router. The feature provides scalable and efficient inline monitoring of flows.

Feature History for Configuring Multicast Routing on the Cisco ASR 9000 Series Routers

Release	Modification
Release 3.9.0	The Video Monitoring feature was introduced.
Release 3.9.1	Included a scenario pertaining to high bandwidth flow support for Video Monitoring service and other related changes.

Contents

Prerequisites for Implemeting Video Monitoring

The following prerequisites are required to implement video monitoring:

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- You must install and activate packages for the advanced video services. For detailed information about optional package installation, see Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide.

- You must install and activate a package for the multicast routing software and enable multicast routing on the system; video monitoring is supported on interfaces that have multicast enabled. For detailed information about multicast routing, refer to the chapter *Implementing Layer-3 Multicast Routing on Cisco ASR 9000 Series Routers*.

Information About Implementing Video Monitoring

To implement Video Monitoring features in this document you must understand the following concepts:

- Introduction to Video Monitoring, page MCC-80
- Key features supported on Video Monitoring, page MCC-81
- Video monitoring Terminology, page MCC-83

Introduction to Video Monitoring

Poor video experience is a major cause for concern for service providers in terms of service costs and loss of revenue. To avoid the service costs of help desk time, NOC (network operation center) troubleshooting resources, and truck rolls, the capability of monitoring video traffic is essential. On the Cisco ASR9000 routers, problems in video flows can be easily diagnosed by video monitoring.

Packet loss is one of the most common causes of video quality degradation. Its impact is more significant on compressed video flows. The video traffic transported through the service provider IP network is (almost always) compressed video – MPEG (or similar) encoded in most cases. Due to the way the compression occurs, the traffic is extremely loss-sensitive. The video is encoded with an independent frame (I-frame) every few seconds, with subsequent frames being deltas from the I-frame. A 3 ms loss of traffic (roughly one IP packet) can result in a viewing degradation for up to 1.2 seconds if the loss was in an I-frame.

Jitter is a key flow characteristic that requires careful buffer provisioning in the end device. The set top box (STB) that displays the media on a screen needs to decode the video in real-time. It buffers the incoming video stream so that it can decode and display the image smoothly. Large network jitter can lead to the buffer underrun or overrun on the STB. This will create a visual artifact or even a "black screen" on the end display depending on how large the jitter is.

End-to-end delay in transmission is not that significant for a broadcast-only application. However, as the video applications get to be more interactive, the end-to-end latency (delay) becomes a critical Quality of Experience (QoE) component. Data Loss is a major contributor for poor QoE.

The 3 main contributors of poor QoE can be summed up as:

- Packet Loss
- Jitter
- Delay

Video Monitoring plays a very significant role in improving the video quality and thus in enhancing the QoE. Video monitoring is implemented on the routers and enables the network operator(s) to measure and track video transport performance on a per-flow basis. The video packets flow through a router. We can look at the packet headers and compute a metric that gives us a measure of the network performance impacting the quality of the video. This information from multiple routers is compared for the same flow to get a clear end-to-end picture of the video issues in the network and the affected flows.

Problems in video flows (and more generally, any streaming flow) can be diagnosed by video monitoring. The purpose of video monitoring is to detect perturbations and anomalies introduced by the network that cause a degraded QoE, i.e. it measures the transport performance for streaming (video) traffic. Encoding errors, audio-video-lag, etc. cause poor QoE. However, these are introduced by the encoding device and not the network. These errors are not monitored.

Key features supported on Video Monitoring

Direct Measurements from Data Plane

Video monitoring plays a significant role in improving the video quality and thus, enhancing the QoE. Video monitoring implemented on Cisco ASR9000 series router enables the network operator to measure and track video transport performance on a per-flow basis in real time. In clear contrast to the conventional traffic monitoring solutions where sampled flows have to be sent to control plane or extra hardware such as dedicated blade on the router, video monitoring on ASR9000 router performs monitoring operation on data plane. This enables video monitoring on ASR9000 to look at forwarded packets in real time to compute a metric that provides a measure of the network performance impacting the quality of the video.

Local Storage and Remote Access

Video monitoring measures packet loss and jitter at wire-speed and stores the collected information on the router so that the network operator can access it via traditional user interface. Furthermore, the performance metrics measured and stored on multiple routers can be accessed via standard SNMP from a remote operation center so that clear end-to-end picture of the same video flow can be composed and analyzed.

Proactive and Reactive Usages

From service provider's perspective, Video monitoring on Cisco ASR9000 serves both reactive and proactive usage. It can be used to verify the video service quality before expanding the service coverage to new customers. Also, it is a powerful analyzing tool to be used in response to service trouble tickets or customer calls. Network operator can configure video monitoring to raise an alarm for various events such as variation in packet loss, jitter, flow rate, number of flows, and etc. Such an alarm can be configured to trigger at any possible value or range.

Flow on Video Monitoring

Video monitoring uses four pieces of packet header fields to distinguish a unique flow - source IP address, destination IP address, source UDP port, and destination UDP port (this implies protocol ID is always UDP).

Multicast vs. Unicast

In the current release, video monitoring monitors the flows with IPv4 multicast destination address in the IP header. It does not monitor the flows with unicast destination addresses.

Flow Rate Types and Protocol Layer

Video monitoring monitors CBR (constant bit rate) flows at IP layer. In other words, Video monitoring can monitor CBR-encoded media stream(s) (e.g., MPEG-2) encapsulated in UDP datagram inside an IPv4 packet. Video monitoring allows user to configure packet rate at IP layer or bit rate at media layer (along with the number and size of media packets).

Metrics

Video monitoring supports both packet loss and jitter metrics that follow MDI (media delivery index, RFC 4445) definition at the IP-UDP level. The MDI metrics are MLR (media loss rate) and DF (delay factor). Video monitoring uses MRV (media rate variation) which is an extension to MDI MLR, i.e., MLR captures only loss while MRV captures both loss and excess. Video monitoring DF is the same as MDI definition, where DF represents one nominal packet inter-arrival time plus monitored MDI jitter. Along with the two key metrics, Video monitoring supports packet count, byte count, packet rate, bit rate, packet size, TTL (time to live) field in IP header, number of flows, raised alarms, and time stamp for various events.

Note The term MDI jitter, is used to differentiate the correctness of DF metric measured by Video monitoring. MDI jitter is measured by comparing the actual packet arrival time against the nominal arrival reference while simple inter-packet jitter is measured by the time difference between two consecutive packet arrivals. The former captures the performance of CBR flow more precisely than the latter.

Number of Flows

In the current release, video monitoring on Cisco ASR9000 series router supports max 1024 flows per NP (network processor). The number of maximum flows per LC (line card) or per system varies depending on the number of NPs on the LC and/or the number of LCs on the system. Per-chassis flow scale depends on the number of NPs on the chassis.

For example, if you have an ASR 9000 series router box with 4 LCs, and if each LC has 8 NPs, per-chassis flow scale goes up to 1K*8*4 = 32K flows per chassis.

High Availability (HA) Features

Video monitoring on ASR9000 supports HA at various levels. It supports process OIR (online insertion and removable), LC OIR, RSP (route switch processor) fail over, and router reload. Configuration will be persistent for all the HA scenarios. Monitored statistics data are preserved at process OIR and RSP FO.

Interface Types and Direction

To activate video monitoring, user has to configure video monitoring service policy on an interface. There are four types of interfaces that the user can attach the video monitoring policy to- main interface, subinterface, ethernet bundle interface, and ethernet bundle subinterface. video monitoring supports only layer 3 interfaces, and not layer 2 interfaces (i.e., L2transport interfaces on an L2VPN bridge domain). video monitoring can be configured only on input direction of the interface.

Flow Rate and DF Precision

ASR9000 video monitoring offers DF metric performance of 1 ms precision. video monitoring supports standard definition (SD) video traffic (mostly compressed) up to 100 Mbps flow rate.

User Interface for Input

Video monitoring supports traditional CLI (command line interface) input for configuration that follows MQC (modular QoS configuration) syntax. User can configure video monitoring by configuring access control list (ACL), class map, and policy map, and it can be activated by attaching the service policy to an interface. In-place policy modification is not supported. Once attached to an interface, the configured service policy can be modified only after detaching from the interface.

User Interface for Output

Video monitoring offers various show/clear commands to user for retrieving the monitored statistics. Please, refer to the *Video Monitoring Commands on Cisco ASR 9000 Series Routers in the Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference* for detailed description of the video monitoring commands.

User can configure TCA (threshold crossing alert) as a part of policy map to enable video monitoring to generate syslog message for various conditions. User can also retrieve standing alarms using show command or via SNMP pull. XML is supported by video monitoring.

Number of Class Maps and Policy Maps

To use video monitoring, user has to configure class map and policy map that will act as a filter to determine which flow to monitor on the data plane. Video monitoring supports a maximum of 1024 class maps per policy-map, and a maximum of 1024 class maps per system. It supports a maximum of 256 policy maps on the system.

Video monitoring Terminology

To implement and configure video monitoring service on ASR9000 series routers, the user needs to understand video monitoring terminology and concepts.

Interval duration and interval updates

Video monitoring monitors all the packets continuously on data plane for a time period, called the interval duration. Statistics are exported periodically at the end of each interval duration which is configured by user. Such exported statistics are defined as interval updates. Status of a video monitoring flow and its transition is described solely by means of these interval updates. Also, all the exported video monitoring flow statistics are stored in terms of interval udpates.

The interval duration is a very vital video monitoring parameter. Video monitoring configuration anchors upon interval duration (i.e., how often to export, how many exports to store, when to delete inactive flows, and etc.) and all the video monitoring functionalities including raising alarm (e.g., for stopped flows, for flows with performance degradation, and etc.) are based on the contents of interval udpates.

Video monitoring flows

A unique video monitoring flow is an instance of packet stream whose header fields match the configured class map (and its associated access control list). A unique flow is local to the interface to which a video monitoring service policy is attached. A video monitoring flow will be composed of a series of stored interval updates. A unique flow that is created on video monitoring after one monitoring interval is call ed a new flow. Thus, a packet stream that lives shorter than one monitoring interval will not be exported as a video monitoring flow, and thus it will not be stored.

Flow stop

If the router stops receiving packets on a monitored flow for one full interval update or longer, the monitored flow is considered as stopped.

Flow resumption

If a stopped video monitoring flow starts receiving packets, a normal interval update will be exported in the next monitoring interval. A resumed flow will have one or more zero intervals followed by a normal interval update.

Flow switchover

A video monitoring flow on an ethernet bundle interface or on an ethernet bundle subinterface may move from one physical member interface to another, i.e., the packet streams stops flowing on one interface and starts flowing on another. This is defined as flow switchover. In such a case, video monitoring treats pre-switchover flow and post-switchover flow as the same if both the interfaces are on the same line card. Otherwise, it will treat them as two different flows.

Flow deletion

If a stopped video monitoring flow continues to export zero intervals for a configured timeout (in terms of the number of monitoring intervals), the flow is considered as dead and is marked for deletion. The actual deletion for all the marked flows will take place after some delay by the periodic (every 150s) sweeping function. Once deleted, all the exported statistics (i.e., series of interval updates including zero intervals) are completely removed from storage.

Implementing Video Monitoring

Configuring Video Monitoring is a 4-step procedure which includes configuring the relevant class-maps, policy maps and binding the video monitoring policy to an interface.

- Creating IPv4 Access Lists, page MCC-84
- Configuring class-map, page MCC-86
- Configuring policy-map, page MCC-88
- Configuring service policy on an interface, page MCC-97

Creating IPv4 Access Lists

Similar to typical IPv4 access list creation and configuration. An example configuration of ACL for video monitoring is presented here for quick reference. For more details, refer to the *Implementing Access lists and Prefix lists chapter of the Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services configuration guide.*

This task configures a standard IPv4 access list.

Standard access lists use source addresses for matching operations.

Note Video Monitoring policy does not allow explicit **deny** statements in ACL configuration. Also, log or log-input is not supported in ACL configuration.

SUMMARY STEPS

1. **configure**
2. **ipv4 access-list** *name*
3. [*sequence-number*] **remark** *remark*
4. [*sequence-number*] **permit udp source [***source-port*] **destination [***destination-port*]
5. Repeat Step 4 as necessary, adding statements by sequence number where you planned. Use the **no** *sequence-number* command to delete an entry.
6. **end**
   or
   commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**ipv4 access-list** *name* **Example:** `RP/0/RSP0/CPU0:router# ipv4 access-list acl_1`	Enters IPv4 access list configuration mode and configures access list acl_1.
Step 3	[*sequence-number*] **remark** *remark* **Example:** `RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 remark Do not allow user1 to telnet out`	(Optional) Allows you to comment about the following **permit** statement in a named access list. • The remark can be up to 255 characters; anything longer is truncated. • Remarks can be configured before or after **permit** statements, but their location should be consistent.
Step 4	[*sequence-number*] **permit udp source** [*source-port*] **destination** [*destination-port*] **Example:** `RP/0/RSP0/CPU0:router(config-ipv4-acl)# 20 permit udp 172.16.0.0/24 eq 5000 host 225.0.0.1 eq 5000`	Specifies one or more conditions allowed. • Put udp as video monitoring only supports udp. • Use the *source* argument to specify the number of network or host from which the packet is being sent. • Use the optional *source-wildcard* argument to specify the wildcard bits to be applied to the source. • Use the *destination* argument to specify the number of network or host to which the packet is being sent. • Use the optional *destination-wildcard* argument to specify the wildcard bits to be applied to the destination.

	Command or Action	Purpose
Step 5	Repeat Step 4 as necessary, adding statements by sequence number where you planned. Use the **no** *sequence-number* command to delete an entry.	Allows you to revise an access list.

Step 6	**end**
	or
	commit
	
	Example:
	`RP/0/RSP0/CPU0:router(config-ipv4-acl)# end`
	or
	`RP/0/RSP0/CPU0:router(config-ipv4-acl)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		– Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		– Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		– Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring class-map

This task sets up the flow classifier. This may match either an individual flow, or may be an aggregate filter matching several flows.

SUMMARY STEPS

1. **configure**
2. **class-map type traffic** *class-map-name*
3. **match access-group ipv4** *acl-name*
4. **end-class-map**
5. **end or commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**class-map type traffic** *class-map-name* **Example:** `RP/0/RSP0/CPU0:router(config)# class-map type traffic class1`	Enters the class-map mode. The class-map type should always be entered as traffic.
Step 3	**match access-group ipv4** *acl-name* **Example:** `RP/0/RSP0/CPU0:router(config-cmap)# match access-group ipv4 acl1`	Enter the ACL to be matched for this class. Only one ACL can be matched per class.
Step 4	**end-class-map** **Example:** `RP/0/RSP0/CPU0:router(config-cmap)# end-class-map`	Completes the class-map configuration.
Step 5	**end** **or** **commit** **Example:** `RP/0/RSP0/CPU0:router(config-cmap)# commit`	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:` – Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. – Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. – Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring policy-map

The policy map for video monitoring is of type, performance-traffic. Only one level of hierarchy is supported for video monitoring policy-maps. This means that no hierarchical policy map configuration is supported for video monitoring.

The policy map configuration for video monitoring has three parts

- Flow parameters configuration – Specifies the different properties of the flow that will be monitored such as interval duration, required history intervals, timeout, etc
- Metric parameters configuration – Specifies the metrics that need to be calculated for the flow that will be monitored.
- React parameters configuration – Specifies the parameters based on which alerts are generated for the flow.

The hierarchy of configuration is, policy-> class->flow. This means that all the parameters specified above will be applied to all the flows that match a particular class in the policy-map. While it is optional to specify flow and react parameters for the flows matching a given class, it is mandatory to specify the metric parameters for the same.

Configuring policy-map with metric parameters

The metric parameters in a policy map can be:

- Layer3 packet rate or
- Media bit rate (by specifying the number of media packet counts and size in the UDP payload).

Note Layer3 packet rate and Media rate have mutually exclusive configuration commands.

The configuration for each case is described in detail below.

Layer3 packet-rate

SUMMARY STEPS

1. **configure**
2. **policy-map type** *performance-traffic policy-map-name*
3. **class type** *traffic class-name*
4. **monitor metric ip-cbr**
5. **rate layer3 packet** *packet-rate* **pps**
6. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**policy-map type** *performance-traffic policy-map-name* **Example:** `RP/0/RSP0/CPU0:router(config)# policy-map type performance-traffic policy1`	Enters the policy-map mode. The policy-map type should always be entered as performance traffic.
Step 3	**class type** *traffic class-name* **Example:** `RP/0/RSP0/CPU0:router(config-pmap)# class type traffic class-name`	Enter the class-map to be matched for this policy. Multiple classes can be specified for a single policy.
Step 4	**monitor metric ip-cbr** **Example:** `RP/0/RSP0/CPU0:router(config- pmap-c)# monitor metric ip-cbr`	Enters the IP-CBR metric monitor submode. **Note** Currently only ip-cbr metric monitoring is supported for video monitoring.

	Command or Action	Purpose
Step 5	**rate layer3 packet** *packet-rate* **pps** **Example:** RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr) # **rate layer3 packet** *packet-rate* **pps**	Specify the IP layer3 packet rate in pps.
Step 6	**end** **or** **commit** **Example:** RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr) # **end** or RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr) # **commit**	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:` – Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. – Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. – Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Media bit-rate

The metric parameters for media bit-rate consist of specifying the media bit rate, media packet count and packet size.

The rate media option enables the user to specify the number of media payload packets (e.g., MPEG-2 datagrams) that will be present in one UDP packet and the size of each such media payload. It is mandatory to specify the media bit rate.There are no defaults for packet count and packet size in IOS XR Software Release 3.9.1. These values needs to be configured.

Note With the media bit rate configured as 1052800 bps, media packet count as 7, and media packet size as 188 bytes, the media packet rate is 100 pps at layer 3. The calculation is: 1052800 / (7 *188*x 8) = 100 pps.

SUMMARY STEPS

1. **configure**
2. **policy-map type** *performance-traffic policy-map-name*
3. **class type** *traffic class-name*

4. **monitor metric ip-cbr**
5. **rate media** *bit-rate* **{bps|kbps|mbps|gbps}**
6. **media packet count in-layer3** *packet-count*
7. **media packet size** *packet-size*
8. **end or commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**policy-map type** *performance-traffic policy-map-name* **Example:** `RP/0/RSP0/CPU0:router(config)# policy-map type performance-traffic policy1`	Enters the policy-map mode. The policy-map type should always be entered as performance traffic.
Step 3	**class type** *traffic class-name* **Example:** `RP/0/RSP0/CPU0:router(config-pmap)# class type traffic class-name`	Enter the class-map to be matched for this policy. Multiple classes can be specified for a single policy.
Step 4	**monitor metric ip-cbr** **Example:** `RP/0/RSP0/CPU0:router(config- pmap-c)# monitor metric ip-cbr`	Enters the IP-CBR metric monitor submode. **Note** Currently only ip-cbr metric monitoring is supported for video monitoring.
Step 5	**rate media** *bit-rate* **{bps|kbps|mbps|gbps}** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# rate media 100 mbps`	Specifies the media bit rate for the flow in bps, kbps, mbps or gbps. The configuration can be committed here. Optional parameters can also be specified. **Note** The default unit of media bit-rate is kbps.
Step 6	**media packet count in-layer3** *packet-count* **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-ipbr)# media packet count in-layer3 10`	Specifies the number of media packets per IP payload.

	Command or Action	Purpose
Step 7	**media packet size packet-size** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# media packet size 188`	Specifies the size in bytes for each media packet in the IP payload.

Step 8	**end**
	or
	commit
	
	Example:
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# end`
	`or`
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		– Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		– Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		– Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring policy-map with flow parameters

The flow parameters in a policy map are optional.

For video monitoring, the data plane continuously monitors the flows and the metrics are exported at the end of every interval. The duration of this interval and the number of such intervals that need to be stored per flow (history) can also be optionally specified by the user. The following per flow parameters can be specified by the user:

- Interval Duration: The time interval at the end of which, metrics are exported. This has to be specified in multiples of 5 (any value between 10 and 300 seconds). The default value is 30 seconds.
- History: The number of such intervals containing flow information (flow id, metrics etc) that needs be stored per flow. This can be any value between 1 and 60. The default value is 10 intervals.
- Timeout: The timeout in multiples of interval duration after which an inactive flow is marked for deletion. This can be any value between 2 and 60. The default value is 0. (Note: the timeout value of 0 has a special meaning that the flow will never be timed out and hence is a static flow).
- Max Flows per class: The maximum number of flows that need to be monitored per class in the policy. This can be any value between 1 and 1024. The default value is 1024 flows per class.

SUMMARY STEPS

1. **configure**
2. **policy-map type** *performance-traffic policy-map-name*
3. **class type** *traffic class-name*
4. **monitor parameters**
5. **{interval duration** *duration*| **flows** *number of flows* **| history** *intervals* **| timeout** *duration***}**
6. **end or commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**policy-map type** *performance-traffic policy-map-name* **Example:** `RP/0/RSP0/CPU0:router(config)# policy-map type performance-traffic policy1`	Enters the policy-map mode. The policy-map type should always be entered as performance traffic.
Step 3	**class type** *traffic class-name* **Example:** `RP/0/RSP0/CPU0:router(config-pmap)# class type traffic class-name`	Enter the class-map to be matched for this policy. Multiple classes can be specified for a single policy.
Step 4	**monitor parameters** **Example:** `RP/0/RSP0/CPU0:router(config- pmap-c)# monitor parameters`	Enters the flow monitor submode.

	Command or Action	Purpose
Step 5	**{interval duration** *duration*| **flows** *number of flows* | **history** *intervals* | **timeout** *duration}* **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-fparm)# interval duration 10`	• Select the **interval duration** option to specify the interval duration per flow; range is 10 to 300 (must be in multiples of 5). The default value is 30 seconds. • Select the **history** option to specify the maximum number of interval data that will be stored per flow. It can be any value between 1 and 60. The deafult value is 10. • Select the **timeout** option to specify the timeout in multiples of the interval duration after which an inactive flow will be marked for deletion. Range is between 2 and 60. The default value is zero, indicating a static flow. • Select the **flows** option to specify the maximum number of flows that can be monitored per class. Range is between 1 and 1024. The default valueis 1024 flows.

Step 6	**end**
	or
	commit
	
	Example:
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# end`
	or
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		– Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		– Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		– Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring policy-map with react parameters

The react parameters in a policy map are optional.

The react parameters are a direct reference for the user to indicate the flow quality. The flow is continuously monitored and at the end of the interval duration, the statistics are examined to check whether the threshold specified by the user for the specific parameter is exceeded. If it is, a syslog alarm is generated on the console. Once the alarm is set, no further syslog notifications are issued for the condition.

- Media Rate variation (MRV): video monitoring reacts and generates an alarm if the MRV statistic of the flow crosses the user specified threshold.
- Delay Factor: video monitoring reacts and generates an alarm if the Delay Factor statistic of the flow crosses the user specified threshold.
- Media-Stop: video monitoring reacts and generates an alarm if a flow stops, i.e, to indicate that no packets were received for the flow during one full monitoring interval.
- Packet-Rate: video monitoring reacts and generates an alarm if the packet rate of the flow crosses the user specified threshold.
- Flow-Count: video monitoring reacts and generates an alarm if the flow count per class crosses the user specified threshold.

SUMMARY STEPS

1. **configure**
2. **policy-map type performance-traffic** *policy-map-name*
3. **class type traffic** *class-name*
4. **react** *react-id* {**mrv | delay-factor | media-stop | packet-rate | flow-count**}
5. **threshold type immediate**
6. **threshold value {ge | gt | le | lt | range}** *limit*
7. **action syslog**
8. **alarm severity {error | critical | alert | emergency}**
9. **alarm type {discrete | grouped }**
10. **end or commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**policy-map type** *performance-traffic policy-map-name* **Example:** `RP/0/RSP0/CPU0:router(config)# policy-map type performance-traffic policy1`	Enters the policy-map mode. The policy-map type should always be entered as performance traffic.
Step 3	**class type** *traffic class-name* **Example:** `RP/0/RSP0/CPU0:router(config-pmap)# class type traffic class-name`	Enter the class-map to be matched for this policy. Multiple classes can be specified for a single policy.
Step 4	**react** *react-id* **{mrv | delay-factor | packet-rate | flow-count | media-stop}** **Example:** `RP/0/RSP0/CPU0:router(config- pmap-c)# react 1 mrv`	Enters the react parameter configuration submode. The react id specified here needs to be unique per class.
Step 5	**threshold type immediate** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-react)# threshold type immediate`	Specifies the trigger type for the threshold. Currently, the available threshold type is immediate.
Step 6	**threshold value {ge | gt | le | lt | range}** *limit* **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-react)# threshold value ge 50`	Specifies the trigger value range for the threshold.
Step 7	**action syslog** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-react)# action syslog`	The **action** keyword specifies the action to be taken if the threshold limit is surpassed. Currently, syslog action is the only option available.
Step 8	**alarm severity {error | critical | alert | emergency}** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-react)# alarm severity critical`	Specifies the alarm severity for syslog.

	Command or Action	Purpose
Step 9	**alarm type {discrete | grouped }** **Example:** `RP/0/RSP0/CPU0:router(config-pmap-c-react)# alarm type discrete`	Specifies the alarm type. Discrete alarm is raised for all the flows that exceed the threshold value. Grouped alarm is raised when a certain number or percentage of the flows exceeds the threshold value.

Step 10	**end**
	or
	commit
	
	Example:
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# end`
	or
	`RP/0/RSP0/CPU0:router(config-pmap-c-ipcbr)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:`
		– Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		– Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		– Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

With reference to the above configuration procedure:

- For the react parameter, **media-stop** , the **threshold-type** and **threshold-value** options are not applicable.
- For the react parameter, **flow-count**, the **alarm-type** option is not applicable.

Configuring service policy on an interface

The configured policy-map must be attached to an interface in ingress direction in order to enable the Video Monitoring service.

For ethernet bundle interface, service policy can be attached to the bundle parent interface only and not to the physical member interfaces. For ethernet bundle subinterfaces, it can be attached to subinterfaces only. For VLAN subinterfaces, service policy cannot be attached to the main interface.

SUMMARY STEPS

1. **configure**
2. **interface** *type interface-path-id*
3. **service-policy type performance-traffic input** *policy-map-name*
4. **end or commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RSP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**interface** *type interface-path-id* **Example:** `RP/0/RSP0/CPU0:router(config)# interface type interface-path-id`	Configures an interface and enters interface configuration mode. • The type argument specifies an interface type. For more information on interface types, use the question mark (?) online help function. • The instance argument specifies either a physical interface instance or a virtual instance. • The naming notation for a physical interface instance is rack/slot/module/port. The slash (/) between values is required as part of the notation. • The number range for a virtual interface instance varies depending on the interface type.

	Command or Action	Purpose
Step 3	**service-policy type performance-traffic input** *policy-name* **Example:** `RP/0/RSP0/CPU0:router(config-if)#` **service-policy type performance-traffic input** *policy1*	Attaches the policy to the interface in the ingress direction.
Step 4	**end** **or** **commit** **Example:** `RP/0/RSP0/CPU0:router(config-if)#` **end** or `RP/0/RSP0/CPU0:router(config-if)#` **commit**	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:` – Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. – Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. – Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuration examples for Implementing Video Monitoring

This section provides configuration examples for different scenarios.

Scenario-1

An ethernet bundle interface has three physical members over which multicast video traffic is flowing at 300 pps per flow. Use video monitoring to monitor all the flows on this ethernet bundle, and raise a critical-level alarm if the per-flow traffic load is over 10 % of expected rate, and raise an error-level alarm if the delay factor is greater than 4 ms. Report the collected statistics every 10 seconds. Keep the reported statistics for 10 minutes as long as the flow is active. Remove flow statistics if no packets are received for 30 seconds.

Example

```
ipv4 access-list sample-acl
```

```
 10 permit udp any any
!
class-map type traffic match-any sample-class
 match access-group ipv4 sample-acl
 end-class-map
!
policy-map type performance-traffic sample-policy
 class type traffic sample-class
  monitor parameters
   interval duration 10
   history 60
   timeout 3
  !
  monitor metric ip-cbr
   rate layer3 packet 300 pps
  !
  react 100 mrv
   threshold type immediate
   threshold value gt 10.00
   action syslog
   alarm severity error
   alarm type discrete
  !
  react 101  delay-factor
   threshold type immediate
   threshold value gt 4.00
   action syslog
   alarm severity error
   alarm type discrete
  !
 !
 end-policy-map
!
interface Bundle-Ether10
 ipv4 address 172.192.1.1 255.255.255.0
 service-policy type performance-traffic input sample-policy
!
interface TenGigE0/6/0/0
 bundle id 10 mode on
!
interface TenGigE0/6/0/1
 bundle id 10 mode on
!
interface TenGigE0/6/0/2
 bundle id 10 mode on
!
```

Scenario-2

A VLAN subinterface is carrying 100 video streams with a common multicast group address of 225.0.0.1 and varying UDP port numbers. The expected packet rate at IP layer is unknown, but the media bit rate is known to be 1052800 bps. The media payload is known to contain MPEG-2 encoded CBR flows and default packetization is used (i.e., in one UDP payload, there are seven MPEG packets where each is 188 bytes long). Do not monitor over 100 flows. Do not timeout and delete any flow even if flow stops, but raise error-level alarm if the percentage of the stopped flows is over 90 %.

Example

```
ipv4 access-list sample-acl
 10 permit udp any host 225.0.0.1
```

```
!
class-map type traffic match-any sample-class
 match access-group ipv4 sample-acl
 end-class-map
!
policy-map type performance-traffic sample-policy
 class type traffic sample-class
  monitor parameters
   flows 100
!
  monitor metric ip-cbr
   rate media 1052800 bps
  !
  react 100 media-stop
   action syslog
   alarm severity error
   alarm type grouped percent 90
  !
 end-policy-map
!
interface GigabitEthernet0/0/0/0
 no shutdown
!
interface GigabitEthernet0/0/0/0.1
 encapsulation dot1q 500
 ipv4 address 172.192.1.1 255.255.255.0
 service-policy type performance-traffic input sample-policy
!
```

Under **monitor metric ip-cbr**, the following two lines need not be configured as they are the defaults:

- media packet count in-layer3 7
- media packet size 188

But if these parameters are different from default values, then they need to be configured.

Scenario-3

A main interface has three groups of multicast streams where the first group has UDP destination port of 1000, the second group has 2000, and the third group has 3000 and 4000. These three groups of streams flow at 100 pps, 200 pps, and 300 pps, respectively. Limit the maximum number of flows in each group to 300 flows and raise the error-level alarm when they reach 90 % of the provisioned flow capacity.

Example

```
ipv4 access-list sample-acl-1
 10 permit udp any any eq 1000
!
ipv4 access-list sample-acl-2
 10 permit udp any any eq 2000
!
ipv4 access-list sample-acl-3
 10 permit udp any any eq 3000
 20 permit udp any any eq 4000
!
class-map type traffic match-any sample-class-1
 match access-group ipv4 sample-acl-1
 end-class-map
!
class-map type traffic match-any sample-class-2
 match access-group ipv4 sample-acl-2
 end-class-map
```

```
!
class-map type traffic match-any sample-class-3
 match access-group ipv4 sample-acl-3
 end-class-map
!
policy-map type performance-traffic sample-policy
 class type traffic sample-class-1
  monitor parameters
   interval duration 10
   history 60
   timeout 3
   flows 300
  !
  monitor metric ip-cbr
   rate layer3 packet 100 pps
  !
  react 100 flow-count
   threshold type immediate
   threshold value gt 270
   action syslog
   alarm severity error
  !
class type traffic sample-class-2
  monitor parameters
   interval duration 10
   history 60
   timeout 3
   flows 300
  !
  monitor metric ip-cbr
   rate layer3 packet 200 pps
  !
  react 100 flow-count
   threshold type immediate
   threshold value gt 270
   action syslog
   alarm severity error
  !
class type traffic sample-class-1
  monitor parameters
   interval duration 10
   history 60
   timeout 3
   flows 300
  !
  monitor metric ip-cbr
   rate layer3 packet 300 pps
  !
  react 100 flow-count
   threshold type immediate
   threshold value gt 270
   action syslog
   alarm severity error
  !
 !
 end-policy-map
!
interface GigabitEthernet0/0/0/0
 ipv4 address 172.192.1.1 255.255.255.0
 service-policy type performance-traffic input sample-policy
!
```

Scenario-4

A 10GE main interface is receiving six high definition (HD) video streams from the digital contents manager (DCM) directly connected to six HD cameras in a sports stadium. Each HD video stream is uncompressed and its bandwidth is as high as 1.611 Gbps at layer 2, which is equivalent to 140625 pps. These six streams are received with multicast groups of 225.0.0.1 through 225.0.0.6 and UDP port number is 5000. Raise a critical-level alarm when the delay factor of any of the flow is above 2 ms or media loss ratio is above 5 %. Use 10 s interval and keep maximum history. Do not monitor more than 6 flows on this interface. Do not time out inactive flows.

Example

```
ipv4 access-list sample-acl
 10 permit udp any eq 5000 225.0.0.0/24 eq 5000
!
class-map type traffic match-any sample-class
 match access-group ipv4 sample-acl
 end-class-map
!
policy-map type performance-traffic sample-policy
 class type traffic sample-class
  monitor parameters
   interval duration 10
   history 60
   flows 6
  !
  monitor metric ip-cbr
   rate layer3 packet 140625 pps
  !
  react 100 mrv
   threshold type immediate
   threshold value gt 5.00
   action syslog
   alarm severity critical
   alarm type discrete
  !
  react 200 delay-factor
   threshold type immediate
   threshold value gt 2.00
   action syslog
   alarm severity critical
   alarm type discrete
  !
 end-policy-map
!
interface TenGigE0/2/0/0
 ipv4 address 172.192.1.1 255.255.255.0
 service-policy type performance-traffic input sample-policy
!
```

Additional References

The following sections provide references related to implementing multicast routing on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Multicast command reference document	*Cisco ASR 9000 Series Aggregation Services Router Multicast Command Reference*
Getting started material	*Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide*
Modular quality of service command reference document	*Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Command Reference*

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC4445	Proposed Media Delivery Index (MDI)

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport