This chapter addresses AVC configuration and includes the following topics:

- Unified Policy CLI, page 3-1
- Metric Producer Parameters, page 3-2
- Reacts, page 3-2
- NetFlow/IPFIX Flow Record, page 3-3
- Configuration Examples, page 3-9

### Unified Policy CLI

From Cisco IOS XE 3.8, monitoring configuration is done using performance-monitor unified monitor and policy.

```
policy-map type performance-monitor <policy-name>
  [no] parameter default account-on-resolution
  class <class-map name>
    flow monitor <monitor-name> [sampler <sampler name>]
    [sampler <sampler name>]
    monitor metric rtp
```

**Usage Guidelines**

- **Support for:**
  - Multiple flow monitors under a class-map.
  - Up to 5 monitors per attached class-map.
  - Up to 256 classes per performance-monitor policy.

- **No support for:**
  - Hierarchical policy.
  - Inline policy.

- Metric producer parameters are optional.
- Account-on-resolution (AOR) configuration causes all classes in the policy-map to work in AOR mode, which delays the action until the class-map results are finalized (the application is determined by NBAR2).

Attach policy to the interface using following command:

```
interface <interface-name>
    service-policy type performance-monitor <policy-name> {input|output}
```

### Metric Producer Parameters

Metric producer-specific parameters are optional and can be defined for each metric producer for each class-map.

**Note**

Cisco IOS XE 3.8 supports only MediaNet-specific parameters.

```
monitor metric rtp
    clock-rate {type-number|type-name|default} rate
    max-dropout number
    max-reorder number
    min-sequential number
    ssrc maximum number
```

### Reacts

The `react` CLI defines the alerts applied to a flow monitor. Applying reacts on the device requires punting the monitor records to the route processor (RP) for alert processing. To avoid the performance reduction of punting the monitor records to the RP, it is preferable when possible to send the monitor records directly to the Management and Reporting system and apply the network alerts in the Management and Reporting system.

```
react <id> [media-stop|mrv|rtp-jitter-average|transport-packets-lost-rate]
```

### NetFlow/IPFIX Flow Monitor

Flow monitor defines monitor parameters, such as record, exporter, and other cache parameters.

```
flow monitor type performance-monitor <monitor-name>
    record <name | vm-default-rtp | vm-default-tcp>
    exporter <exporter-name>
    history size <size> [timeout <interval>]
    cache entries <num>
    cache timeout {{active | inactive | synchronized} <value> | event transaction end}
    cache type {permanent | normal | immediate}
    react-map <react-map-name>
```

### Usage Guidelines

- The `react-map` CLI is allowed under the class in the policy-map. In this case, the monitor must include the exporting of the class-id in the flow record. The route processor (RP) correlates the class-id in the monitor with the class-id where the react is configured.
- Applying history or a react requires punting the record to the RP.
• Export on the “event transaction end” is used to export the records when the connection or transaction is terminated. In this case, the records are not exported based on timeout. Exporting on the event transaction end should be used when detailed connection/transaction granularity is required, and has the following advantages:
  – Sends the record close to the time that it has ended.
  – Exports only one record on true termination.
  – Conserves memory in the cache and reduces the load on the Management and Reporting system.
  – Enables exporting multiple transactions of the same flow. (This requires a protocol pack that supports multi-transaction.)

NetFlow/IPFIX Flow Record

The flow record defines the record fields. With each Cisco IOS release, the Cisco AVC solution supports a more extensive set of metrics.

The sections that follow list commonly used AVC-specific fields as of release IOS XE 3.8, organized by functional groups. These sections do not provide detailed command reference information, but highlight important usage guidelines.

In addition to the fields described below, a record can include any NetFlow field supported by the ASR 1000 platform.


Note

In this release, the record size is limited to 30 fields (key and non-key fields or match and collect fields).

L3/L4 Fields

The following are L3/L4 fields commonly used by the Cisco AVC solution.

```
[collect | match] connection [client|server] [ipv4|ipv6] address
[collect | match] connection [client|server] transport port
[collect | match] [ipv4|ipv6] [source|destination] address
[collect | match] transport [source-port|destination-port]
[collect | match] [ipv4|ipv6] version
[collect | match] [ipv4|ipv6] protocol
[collect | match] routing vrf [input|output]
[collect | match] [ipv4|ipv6] dscp
[collect | match] ipv4 ttl
[collect | match] ipv6 hop-limit
collect transport tcp option map
collect transport tcp window-size [minimum|maxumimum|sum]
collect transport tcp maximum-segment-size
```

Usage Guidelines

The client is determined according to the initiator of the connection.

The **client** and **server** fields are bi-directional. The **source** and **destination** fields are uni-directional.
L7 Fields

The following are L7 fields commonly used by the Cisco AVC solution.

\[
\text{[collect | match] application name [account-on-resolution]}
\]
collect application http url
collect application http host
collect application http user-agent
collect application http referer
collect application rtsp host-name
collect application smtp server
collect application smtp sender
collect application pop3 server
collect application nntp group-name
collect application sip source
collect application sip destination

Usage Guidelines

- The application ID is exported according to RFC-6759.
- Account-On-Resolution configures FNF to collect data in a temporary memory location until the record key fields are resolved. After resolution of the record key fields, FNF combines the temporary data collected with the standard FNF records. Use the `account-on-resolution` option when the field used as a key is not available at the time that FNF receives the first packet.

The following limitations apply when using Account-On-Resolution:

- Flows ended before resolution are not reported.
- FNF packet/octet counters, timestamp, and TCP performance metrics are collected until resolution. All other field values are taken from the packet that provides resolution or the following packets.

- For information about extracted fields, including the formats in which they are exported, see Appendix B, “DPI/L7 Extracted Fields”.

Interfaces and Directions

The following are interface and direction fields commonly used by the Cisco AVC solution:

\[
\text{[collect | match] interface [input|output]}
\]
\[
\text{[collect | match] flow direction}
\]
collect connection initiator

Counters and Timers

The following are counter and timer fields commonly used by the Cisco AVC solution:

collect connection client counter bytes [long]
collect connection client counter packets [long]
collect connection server counter bytes [long]
collect connection server counter packets [long]
collect counter packets [long]
collect counter bytes [long]
collect counter bytes rate
collect connection server counter responses
collect connection client counter packets retransmitted
collect connection transaction duration (sum, min, max)
collect connection transaction counter complete
TCP Performance Metrics

The following are fields commonly used for TCP performance metrics by the Cisco AVC solution:

- collect connection delay network to-server \{sum, min, max\}
- collect connection delay network to-client \{sum, min, max\}
- collect connection delay network client-to-server \{sum, min, max\}
- collect connection delay response to-server \{sum, min, max\}
- collect connection delay response to-server histogram [bucket1 ... bucket7 | late]
- collect connection delay response client-to-server \{sum, min, max\}
- collect connection delay application \{sum, min, max\}

Usage Guidelines

The following limitations apply to TCP performance metrics in AVC for IOS XE 3.8:

- All TCP performance metrics must observe bi-directional traffic.
- The policy-map must be applied in both directions.

**Figure 3-1** provides an overview of network response time metrics.

**Figure 3-1** Network response times
Figure 3-2 provides details of network response time metrics.

**Figure 3-2     Network response time metrics in detail**

![Network response time metrics diagram]

**Media Performance Metrics**

The following are fields commonly used for media performance metrics by the Cisco AVC solution:

- `[collect | match] match transport rtp ssrc`
- `collect transport rtp payload-type`
- `collect transport rtp jitter mean sum`
- `collect transport rtp jitter [minimum | maximum]`
- `collect transport packets lost counter`
- `collect transport packets expected counter`
- `collect transport packets lost counter`
- `collect transport packets lost rate`
- `collect transport event packet-loss counter`
- `collect counter packets dropped`
- `collect application media bytes counter`
- `collect application media bytes rate`
- `collect application media packets counter`
- `collect application media packets rate`
- `collect application media event`
- `collect monitor event`
Usage Guidelines

Some of the media performance fields require punt to the route processor (RP). For more information, see Appendix C, “Fields that Require Punt to the Route Processor”.

L2 Information

The following are L2 fields commonly used by the Cisco AVC solution:

\[\text{collect | match] datalink [source-vlan-id | destination-vlan-id]}\]
\[\text{collect | match] datalink mac [source | destination] address [input | output]}\]

WAAS Interoperability

The following are WAAS fields commonly used by the Cisco AVC solution:

\[\text{collect | match] services waas segment [account-on-resolution]}\]
\[\text{collect services waas passthrough-reason}\]

Usage Guidelines

Account-On-Resolution configures FNF to collect data in a temporary memory location until the record key fields are resolved. After resolution of the record key fields, FNF combines the temporary data collected with the standard FNF records. Use this option (account-on-resolution) when the field used as a key is not available at the time that FNF receives the first packet.

The following limitations apply when using Account-On-Resolution:

- Flows ended before resolution are not reported.
- FNF packet/octet counters, timestamp and TCP performance metrics are collected until resolution.
  All other field values are taken from the packet that provides resolution or the following packets.

Classification

The following are classification fields commonly used by the Cisco AVC solution:

\[\text{collect | match] policy performance-monitor classification hierarchy}\]

Usage Guidelines

Use this field to report the matched class for the performance-monitor policy-map.

Connection/Transaction Metrics

The following are connection/transaction metrics fields commonly used by the Cisco AVC solution:

\[\text{collect | match] connection transaction-id}\]
\[\text{collect flow sampler}\]

Usage Guidelines

In IOS XE 3.8, transaction-id reports a unique value for each connection.
NetFlow/IPFIX Option Templates

NetFlow option templates map IDs to string names and descriptions:

```
flow exporter my-exporter
export-protocol ipfix
    template data timeout <timeout>
    option interface-table timeout <timeout>
    option vrf-table timeout <timeout>
    option sampler-table timeout <timeout>
    option application-table timeout <timeout>
    option application-attributes timeout <timeout>
    option sub-application-table timeout <timeout>
    option c3pl-class-table timeout <timeout>
    option c3pl-policy-table timeout <timeout>
```

NetFlow/IPFIX Show commands

Use the following commands to show or debug NetFlow/IPFIX information:

```
show flow monitor type performance-monitor [<name> [cache [raw]]]
show flow record type performance-monitor
show policy-map type performance-monitor [<name> | interface]
```

NBAR Attribute Customization

Use the following commands to customize the NBAR attributes:

```
[no] ip nbar attribute-map <profile name>
    attribute category <category>
    attribute sub-category <sub-category>
    attribute application-group <application-group>
    attribute tunnel <tunnel-info>
    attribute encrypted <encrypted-info>
    attribute p2p-technology <p2p-technology-info>
[no] ip nbar attribute-set <protocol-name> <profile name>
```

Note

These commands support all attributes defined by the NBAR2 Protocol Pack, including custom-category, custom-sub-category, and custom-group available in Protocol Pack 3.1.

NBAR Customize Protocols

Use the following commands to customize NBAR protocols and assign a protocol ID. A protocol can be matched based on HTTP URL/Host or other parameters:

```
ip nbar custom <protocol-name> [http {{url <urlregexp>}} [host <hostregexp>]}] [offset [format value]] [variable field-name field-length] [source | destination] [tcp | udp ] [range start end | port-number ] [id <id>]
```
Packet Capture Configuration

Use the following commands to enable packet capture:

```
policy-map type packet-services <policy-name>
  class <class-name>
    capture limit packet-per-sec <pps> allow-nth-pak <np> duration <duration>
    packets <packets> packet-length <len>
    buffer size <size> type <type>

interface <interface-name>
  service-policy type packet-services <policy-name> [input|output]
```

Configuration Examples

This section contains configuration examples for the Cisco AVC solution. These examples provide a general view of a variety of configuration scenarios. Configuration is flexible and supports different types of record configurations.

Conversation Based Records—Omitting the Source Port

The monitor configured in the following example sends traffic reports based on conversation aggregation. For performance and scale reasons, it is preferable to send TCP performance metrics only for traffic that requires TCP performance measurements. It is recommended to configure two similar monitors:

- One monitor includes the required TCP performance metrics. In place of the line shown in bold in the example below (collect <any TCP performance metric>), include a line for each TCP metric for the monitor to collect.
- One monitor does not include TCP performance metrics.

The configuration is for IPv4 traffic. Similar monitors should be configured for IPv6.

```
flow record type performance-monitor conversation-record
  match services waas segment account-on-resolution
  match connection client ipv4 (or ipv6) address
  match connection server ipv4 (or ipv6) address
  match connection server transport port
  match ipv4 (or ipv6) protocol
  match application name account-on-resolution
  collect interface input
  collect interface output
  collect connection server counter bytes long
  collect connection client counter bytes long
  collect connection server counter packets long
  collect connection client counter packets long
  collect connection sum-duration
  collect connection new-connections
  collect policy qos class hierarchy
  collect policy qos queue id
  collect <any TCP performance metric>
```

```
flow monitor type performance-monitor conversation-monitor
  record conversation-record
  exporter my-exporter
  history size 0
```
cache type synchronized
cache timeout synchronized 60
cache entries <cache size>

HTTP URL

The monitor configured in the following example sends the HTTP host and URL. If the URL is not required, the host can be sent as part of the conversation record (see Conversation Based Records—Omitting the Source Port, page 3-9).

flow record type performance-monitor url-record
    match transaction-id
    collect application name
    collect connection client ipv4 (or ipv6) address
    collect routing vrf input
    collect application http url
    collect application http host
    <other metrics could be added here if needed.
    For example bytes/packets to calculate BW per URL
    Or performance metrics per URL>

flow monitor type url-monitor
    record url-record
    exporter my-exporter
    history size 0
    cache type normal
    cache timeout event transaction-end
    cache entries <cache size>

Application Traffic Statistics

The monitor configured in the following example collects application traffic statistics:

flow record type performance-monitor application-traffic-stats
    match ipv4 protocol
    match application name account-on-resolution
    match ipv4 version
    match flow direction
    collect connection initiator
    collect counter packets
    collect counter bytes long
    collect connection new-connections
    collect connection sum-duration

flow monitor type application-traffic-stats
    record application-traffic-stats
    exporter my-exporter
    history size 0
    cache type synchronized
    cache timeout synchronized 60
    cache entries <cache size>
Media RTP Report

The monitor configured in the following example reports on media traffic:

```text
flow record type performance-monitor media-record
   match ipv4(or ipv6) protocol
   match ipv4(or ipv6) source address
   match ipv4(or ipv6) destination address
   match transport source-port
   match transport destination-port
   match transport rtp ssrc
   match routing vrf input
   collect transport rtp payload-type
   collect application name
   collect counter packets long
   collect counter bytes long
   collect transport rtp jitter mean sum
   collect transport rtp payload-type
   collect <other media metrics>

flow monitor type media-monitor
   record media-record
   exporter my-exporter
   history size 10 // default history
   cache type synchronized
   cache timeout synchronized 60
   cache entries <cache size>
```

Policy-Map Configuration and Applying to an Interface

The following example illustrates how to configure a reporting policy-map and apply it to an interface.

- The classes definition is not shown.
- Media report is not included.

```text
policy-map type performance-monitoring my-policy
   parameter default account-on-resolution
   class ip_tcp_http
      monitor url_monitor
      monitor conversation-monitor
   class ip_tcp_art
      monitor conversation-monitor
   class ip_tcp_udp-rest
      monitor application-traffic-stats

platform qos performance-monitor

interface GigabitEthernet0/0/0
service-policy type performance-monitor input my-policy
service-policy type performance-monitor output my-policy
```
Control and Throttle Traffic

Use the following to control and throttle the peer-to-peer (P2P) traffic in the network to 1 megabit per second:

```plaintext
class-map match-all p2p-class-map
    match protocol attribute sub-category p2p-file-transfer

policy-map p2p-attribute-policy
    class p2p-class-map
        police 1000000
    Int Gig0/0/3
        service-policy input p2p-attribute-policy
```