System Management Configuration Guide, Cisco Catalyst IE9300 Rugged Series Switches

Cisco Catalyst IE9300 Rugged Series
Updated September 28, 2026

Topics included

1 AVC in a Wired Network....4
Application Visibility and Control....5
AVC class map and policy map formats....5
Guidelines and limitations for wired AVC....7
Configure AVC on wired ports....9
Enable application recognition on an interface....9
Create an AVC QoS policy....10
Create a Class Map....10
Create a policy map....11
Apply QoS policy to the switch port....12
Configure Wired AVC Flexible NetFlow....13
Create a flow record....13
Configure a bidirectional flow record for record-1....14
Configure a bidirectional flow record for record-2....17
Configure a flow record for ingress traffic....21
Configure a flow record for egress traffic....23
Configure a flow monitor....26
Associate flow monitor to an interface....28
NBAR2 custom applications....29
HTTP customization....30
SSL customization....30
DNS customization....30
Composite customization....31
L3/L4 Customization....31
Monitor custom applications....31
NBAR2 dynamic hitless protocol pack....32
Load NBAR2 protocol pack....32
Monitor AVC statistics on the switch....33
AVC configuration examples....34
AVC troubleshooting: questions and answers....44
Feature history for AVC in a wired network....45

2 Swap Drive....46
Swap Drive....47
Guidelines for using SD cards and USB drives....47
How Swap Drive works....48
Swap the external drive....48
Swap Drive commands....49

3 Dying Gasp....52
Dying Gasp....53
Enable dying-gasp notification....53

4 Factory Reset....56
Factory reset procedures....57
Restrictions for performing a factory reset....58
Secure data wipes....58
Configure factory reset....58
Feature history for factory reset....61

5 Migration to Meraki Cloud Management Dashboard....62
Meraki cloud management dashboard migration....63
Supported switch models for cloud-managed mode....63
How to migrate to a Meraki cloud management dashboard....64
Configure the device clock....65
DHCP IP and DNS configuration....66
Validate device compatibility....67
Register and migrate to Meraki cloud management dashboard....68
Revert the switch to IOS-XE mode....69
How the device behaves during migration....69

1 AVC in a Wired Network

Topics:

- Application Visibility and Control
- AVC class map and policy map formats
- Guidelines and limitations for wired AVC
- Configure AVC on wired ports
- Create an AVC QoS policy
- Configure Wired AVC Flexible NetFlow
- NBAR2 custom applications
- NBAR2 dynamic hitless protocol pack
- Monitor AVC statistics on the switch
- AVC troubleshooting: questions and answers
- Feature history for AVC in a wired network

Application Visibility and Control

This topic explains Application Visibility and Control (AVC), a Cisco solution that enables application-aware and application-intelligent network operations by classifying applications using deep packet inspection and providing detailed application statistics.

Application Visibility and Control (AVC) is a Cisco solution that classifies applications using deep packet inspection with the Network-Based Application Recognition (NBAR2) engine, enabling application-aware and application-intelligent network operations.

AVC classification and monitoring

AVC enables application-aware network operations by classifying and monitoring application traffic using advanced inspection and reporting tools. It achieves this visibility through these configuration methods.

These are the key classification and monitoring capabilities of AVC:

- AVC classifies applications using Network-Based Application Recognition (NBAR2) deep packet inspection.
- You can configure AVC on wired access ports for standalone switches.
- You can enable NBAR2 can be enabled explicitly using protocol-discovery or implicitly by attaching a QoS policy with a **match protocol** classifier.
- You can configure wired AVC Flexible NetFlow (FNF) on an interface to provide per-interface statistics for clients, servers, and applications.
- Traffic monitoring is similar to **application-client-server-stats** in **application-statistics** and **application-performance** profiles in Easy Performance Monitor (ezPM).

AVC class map and policy map formats

This topic explains the supported AVC class map and policy map formats, including examples and directionality for ingress and egress.

AVC class map formats

Class Map Format	Class map example	Direction
match protocol *protocol name*	`class-map match-any NBAR-VOICE` ` match protocol ms-lync-audio`	Both ingress and egress

Combination filters	`class-map match-any NBAR-VOICE`
	`match protocol ms-lync-audio`
	`match dscp ef`
		Both ingress and egress

AVC policy formats

Policy format	QoS action
Egress policy based on match protocol filter	Mark and police
Ingress policy based on match protocol filter	Mark and police

This table describes AVC policy formats with examples.

AVC Policy Format	AVC Policy Example

Basic set	`policy-map MARKING-IN`
	`class NBAR-MM_CONFERENCING`
	`set dscp af41`

Basic police	`policy-map POLICING-IN`
	`class NBAR-MM_CONFERENCING`
	`police cir 600000`
	`set dscp af41`

Basic set and police	`policy-map webex-policy`
	`class webex-class`
	`set dscp ef`
	`police 5000000`

Multiple set and police including default	`policy-map webex-policy`
	`class webex-class`
	`set dscp af31`
	`police 4000000`
	`class class-webex-category`
	`set dscp ef`
	`police 6000000`
	`class class-default`
	`set dscp <>`

Hierarchical police	`policy-map webex-policy`
	`class webex-class`
	`police 5000000`
	`service-policy client-in-police-only`
	`policy-map client-in-police-only`
	`class webex-class`
	`police 100000`
	`class class-webex-category`
	`set dscp ef`
	`police 200000`

Hierarchical set and police	`policy-map webex-policy`
	`class class-default`
	`police 1500000`
	`service policy client-up-child`
	`policy-map client-up-child`
	`class webex-class`
	`police 100000`
	`set dscp ef`
	`class class-webex-category`
	`police 200000`
	`set dscp af31`

Guidelines and limitations for wired AVC

Use these guidelines when configuring Wired AVC features to ensure compatibility, supported interface types, and feature

limitations across different platforms and interface configurations.

These are NBAR-based QoS policy configuration guidelines.

Interface and protocol

- Attachment should be done only on physical layer 2 and layer 3 ports, and these ports cannot be part of a port channel. Attachment to trunk ports are not supported.
- Do not configure NBAR with transmit (Tx) Switched Port Analyzer (SPAN) on the same interface.
- Use only IPv4 unicast (TCP/UDP) traffic.
- IPv6 classification is not supported.
- Use NBAR2 match criteria (**match protocol**) only with marking or policing actions. Do not use NBAR2 match criteria with queuing features. Limit concurrent protocols to 255 across all policies due to hardware constraints in **match protocol**.
- Do not configure AVC on
  - Logical interfaces.
  - The management port (Gig 0/0).

Feature compatibility

- Do not configure NBAR and ACL logging on the same switch.
- Attach only one NBAR-based QoS mechanism (either protocol-based or attribute-based) to a port at a time.
- Only these attributes are supported for attribute-based QoS
  - `traffic-class`
  - `business-relevance`
- Protocol-discovery, application-based QoS, and wired AVC Flexible NetFlow (FNF) can be configured together, but not with non-application-based FNF on the same interface at the same time.
- You can attach up to two wired AVC monitors (with different predefined records) per interface.

Management and performance

- The Web UI supports application visibility configuration and monitoring. Use the CLI to perform application control for NBAR-based QoS. Web UI does not support Application Control.
- To manage and check wired AVC traffic from the Web UI, first configure these commands using the CLI:
  - **ip http authentication local**
  - **ip nbar http-service**

Legacy WDAVC QoS limitations

- Only marking and policing are supported.
- Only physical interfaces are supported.
- There is a delay in the QoS classification since the application classification is done offline (while the initial packet/s of the flow are meanwhile forwarded before the correct QoS classification).

Configure AVC on wired ports

Learn how to configure application visibility and control (AVC) on wired ports by enabling protocol-discovery, creating and applying AVC QoS policies, and configuring application-based Flexible NetFlow.

This task guides you through configuring AVC on wired ports, including protocol-discovery, AVC QoS policies, and application-based flexible NetFlow.

Protocol-Discovery, application-based QoS, and application-based flexible NetFlow are independent features. You can configure them independently or together on the same interface at the same time.

Procedure

1. Configure visibility. Enable protocol-discovery on the interface.

   Activate NBAR2 engine by enabling protocol-discovery using the **ip nbar protocol-discovery** command in interface configuration mode. For more details, refer to Enable application recognition on an interface on page 9
2. Create AVC QoS policy
3. Configure application-based Flexible Netflow

Enable application recognition on an interface

Learn how to enable application recognition on an interface using the CLI by configuring protocol-discovery with NBAR2 engine.

To enable application recognition on an interface, complete the following steps:

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```
2. Use the **interface** *interface-id* command to specify the interface for which protocol-discovery is enabled.

   Example

   ```
   Switch(config)# interface gigabitethernet 1/1
   ```
3. Use the **ip nbar protocol-discovery** command to enable application recognition on the interface by activating NBAR2 engine.

   Example

   ```
   Switch(config-if)# ip nbar protocol-discovery
   ```
4. Use the **end** command to return to privileged EXEC mode.

Example

```
Switch(config-if)# end
```

Create an AVC QoS policy

Learn how to create an AVC Quality of Service (QoS) policy by defining a class map, creating a policy map, and applying the QoS policy to a switch port.

Perform these tasks to create an AVC Quality of Service (QoS) policy.

This procedure is applicable to IE9300 switches and is cloned from the 9200 documentation.

Procedure

1. Create a class map with match protocol filters .
2. Create a policy map
3. Apply a QoS policy to the switch port

Create a Class Map

Learn how to create a class map before configuring any match protocol filter. Use this procedure to apply QoS actions such as marking and policing to traffic on wired access ports.

You must create a class map before configuring any match protocol filter. You can apply the QoS actions such as marking and policing to the traffic. The AVC match protocol filters are applied to the wired access ports. For more information about the protocols that are supported, refer to http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/prot_lib/config_library/nbar-prot-pack-library.html .

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **class-map** *class-map-name* command to create a class map.

   Example

   ```
   Switch(config)# class-map webex-class
   ```

3. Use the **match protocol** *application-name* command to specify a match to the application name.

   Example

   ```
   Switch(config-cmap)# match protocol webex-media
   ```

4. Use the **end** command to return to the privileged EXEC mode.

Example

```
Switch(config-if)# end
```

Create a policy map

Learn how to create a policy map on the IE9300 switch by configuring global mode, defining a policy map, classifying traffic, configuring a policer, and setting traffic classification parameters.

Complete the following steps to create a policy map.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

Example

```
Switch# configure terminal
```

2. Use the **policy-map** *policy-map-name* command to create a policy map.

Example

```
Switch(config)# policy-map webex-policy
```

By default, no policy maps are defined.

The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.

> **Note**
>
> To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command.

3. Use the **class** [*class-map-name* | **class-default**] command todefine a traffic classification.

Example

```
Switch(config-pmap)# class webex-class
```

By default, no policy map and class maps are defined.

If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.

A **class-default** traffic class is predefined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** is included in the **class-default** class, all packets that have not already matched the other traffic classes will match **class-default** .

> **Note**
>
> To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command.

4. Use the **police** *rate-bps burst-byte* command to configure a policer.

   Example

   ```
   Switch(config-pmap-c)# police 100000 80000
   ```

   Defines a policer for the classified traffic.

   By default, no policer is defined.

   - For *rate-bps,* specify an average traffic rate in bits per second (b/s). The range is 8000 to 10000000000.
   - For *burst-byte,* specify the normal burst size in bytes. The range is 1000 to 512000000.

5. Use the **set dscp** *new-dscp* **cos** *cos-value* command to classify the IP traffic.

   Example

   ```
   Switch(config-pmap-c)# set dscp 45
   ```

   Classifies IP traffic by setting a new value in the packet.

   - For **dscp** *new-dscp* , enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63.

6. Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

Apply QoS policy to the switch port

Learn how to apply a QoS policy to a switch port by entering configuration mode, selecting the interface, applying the policy, and returning to privileged EXEC mode.

Complete the following steps to apply a QoS policy to the switch port.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **interface** *interface-id* command to enter interface configuration mode.

   Example

   ```
   Switch(config)# interface Gigabitethernet 1/1
   ```

3. Use the **service-policy input** *policymapname* command to apply the local policy to the interface.

   Example

   ```
   Switch(config-if)# service-policy input MARKING_IN
   ```

4. Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

Configure Wired AVC Flexible NetFlow

Learn how to configure Wired AVC Flexible NetFlow by performing a series of tasks including creating flow records, exporters, and associating flow monitors to interfaces.

Perform these tasks to configure Wired AVC Flexible NetFlow.

Procedure

1. Create a flow record
2. Configure a bidirectional flow record for record-1 on page 14
3. Configure a bidirectional flow record for record-2 on page 17
4. Configure a flow record for ingress traffic on page 21
5. Configure a flow record for egress traffic on page 23
6. Create a flow exporter
7. Associate a flow monitor to an interface

Create a flow record

Wired AVC Flexible Netflow (FNF) supports two types of predefined flow records–legacy bidirectional flow records and directional flow records (ingress and egress).

You can configure the following predefined flow records and associate them with a flow monitor:

- Two bidirectional flow records
- Two directional flow records

The legacy bidirectional records are client/server application statistics records, and the new directional records are application-stats for input/output.

Configure a bidirectional flow record for record-1

Learn how to create a bidirectional flow record on Cisco IE9300 Switchs by configuring flow record parameters, matching criteria, and collecting relevant statistics for AVC support.

Use this procedure to create a bidirectional flow record. This task enables you to configure flow record parameters and collect statistics for AVC support on Cisco IE9300 switches.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **flow record** *flow_record_name* command to enter flow record configuration mode

   Example

   ```
   Switch(config)# flow record fr-wdavc-1
   ```

3. (Optional) Use the **description** *description* command to create a description for the flow record.

   Example

   ```
   Switch(config-flow-record)# description fr-wdavc-1
   ```

4. Use these command to configure match criteria for IP version, protocol, and application name.
   a) Use the **match ipv4 version** command to specify a match to the IP version from the IPv4 header.

      Example

      ```
      Switch(config-flow-record)# match ipv4 version
      ```

   b) Use the **match ipv4 protocol** command to specify a match to the IPv4 protocol.

      Example

      ```
      Switch(config-flow-record)# match ipv4 protocol
      ```

   c) Use the **match application name** command to specify a match to the application name.

      Example

      ```
      Switch(config-flow-record)# match application name1
      ```

> **Note**
>
> This action is mandatory for AVC support, as the action allows the flow to be matched against the application.

5. Use these commands to specify match criteria for the client and server IPv4 addresses, server transport port, and flow observation point.
   a) Use the **match connection client ipv4 address** command to specify a match to the IPv4 address of the client (flow initiator).

      Example

      ```
      Switch(config-flow-record)# match connection client ipv4 address
      ```

   b) Use the **match connection client transport port** command to specify a match to the transport port of the server.

      Example

      ```
      Switch(config-flow-record)# match connection client transport port
      ```

   c) Use the **match connection server ipv4 address** command to specify a match to the IPv4 address of the server (flow responder).

      Example

      ```
      Switch(config-flow-record)# match connection server ipv4 address
      ```

   d) Use the **match connection server transport port** command to specify a match to the transport port of the server.

      Example

      ```
      Switch(config-flow-record)# match connection server transport port
      ```

   e) Use the **match flow observation point** command to specify a match to the observation point ID for flow observation metrics.

      Example

      ```
      Switch(config-flow-record)# match flow observation point
      ```

6. Use these commands to configure collection of flow direction and connection initiator.
   a) Use the **collect flow direction** command to specify the collector flow direction.

      Example

      ```
      Switch(config-flow-record)# collect flow direction
      ```

      Specifies to collect the direction – Ingress or Egress – of the relevant side – Initiator or Responder – of the bi-directional flow that is specified by the **initiator** keyword in the **collect connection initiator** command in the following step. Depending on the value specified by the **initiator** keyword, the **flow direction** keyword takes the following values:

- 0x01 = Ingress Flow
- 0x02 = Egress Flow

When the **initiator** keyword is set to initiator, the flow direction is specified from the initiator side of the flow. When the initiator keyword is set to responder, the flow direction is specified from the responder side of the flow. For wired AVC, the **initiator** keyword is always set to initiator.

b) Use the **collect connection initiator** command to specify the collector side of the flow (initiator or responder).

Example

```
Switch(config-flow-record)# collect connection initiator
```

Direction of the flow specified – **collect flow direction** command.

The **initiator** keyword provides the information about the direction of the flow:

- 0x01 = Initiator - the flow source is the initiator of the connection.

For wired AVC, the **initiator** keyword is always set to initiator.

7. Use these commands to collect connection initiation and traffic counters for client and server.

a) Use the **collect connection new-connections** command to specify the number of connection initiations observed.

Example

```
Switch(config-flow-record)# collect connection new-connections
```

b) Use the **collect connection client counter packets long** command to specify the number of packets sent by the client.

Example

```
Switch(config-flow-record)# collect connection client counter packets long
```

c) Use the **collect connection client counter bytes network long** command to specify the total number of bytes transmitted by the client.

Example

```
Switch(config-flow-record)# collect connection client counter bytes network
 long
```

d) Use the **collect connection server counter packets long** command to specify the number of packets sent by the server.

Example

```
Switch(config-flow-record)# collect connection server counter packets long
```

e) Use the **collect connection server counter bytes network long** command to specify the the total number of bytes transmitted by the server.

Example

```
Switch(config-flow-record)# collect connection server counter bytes network
 long
```

8. Use these command to collect absolute timestamps for the first and last packets in the flow.
   a) Use the **collect timestamp absolute first** command to specify the time, in milliseconds, when the first packet was seen in the flow.

   Example

   ```
   Switch(config-flow-record)# collect timestamp absolute first
   ```

   b) Use the **collect timestamp absolute last** command to specify the time, in milliseconds, when the most recent packet was seen in the flow.

   Example

   ```
   Switch(config-flow-record)# collect timestamp absolute last
   ```

9. Use these commands to Exit configuration mode and verify the flow record.
   a) Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

   b) Use the **show flow record** command to display information about all the flow records.

   Example

   ```
   Switch# show flow record
   ```

Configure a bidirectional flow record for record-2

Learn how to create a bidirectional flow record on Cisco IE9300 Switchs by configuring flow record parameters, matching criteria, and collecting relevant statistics for AVC support.

Use this procedure to create a bidirectional flow record. This task enables you to configure flow record parameters and collect statistics for AVC support on Cisco IE9300 switches.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **flow record** *flow_record_name* command to enter flow record configuration mode

Example

```
Switch(config)# flow record fr-wdavc-1
```

3. (Optional) Use the **description** *description* command to create a description for the flow record.

   Example

   ```
   Switch(config-flow-record)# description fr-wdavc-1
   ```

4. Use these command to configure match criteria for IP version, protocol, and application name.
   a) Use the **match ipv4 version** command to specify a match to the IP version from the IPv4 header.

      Example

      ```
      Switch(config-flow-record)# match ipv4 version
      ```

   b) Use the **match ipv4 protocol** command to specify a match to the IPv4 protocol.

      Example

      ```
      Switch(config-flow-record)# match ipv4 protocol
      ```

   c) Use the **match application name** command to specify a match to the application name.

      Example

      ```
      Switch(config-flow-record)# match application name1
      ```

      Note

      This action is mandatory for AVC support, as the action allows the flow to be matched against the application.

5. Use these commands to specify match criteria for the client and server IPv4 addresses, server transport port, and flow observation point.
   a) Use the **match connection client ipv4 address** command to specify a match to the IPv4 address of the client (flow initiator).

      Example

      ```
      Switch(config-flow-record)# match connection client ipv4 address
      ```

   b) (Optional) Use the **match connection client transport port** command to specify a match to the connection port of the client as a key field for a flow record.

      Example

      ```
      Switch(config-flow-record)# match connection client transport port
      ```

c) Use the **match connection server ipv4 address** command to specify a match to the IPv4 address of the server (flow responder).

Example

```
Switch(config-flow-record)# match connection server ipv4 address
```

d) Use the **match connection server transport port** command to specify a match to the transport port of the server.

Example

```
Switch(config-flow-record)# match connection server transport port
```

e) Use the **match flow observation point** command to specify a match to the observation point ID for flow observation metrics.

Example

```
Switch(config-flow-record)# match flow observation point
```

6. Use these commands to configure collection of flow direction and connection initiator.

a) Use the **collect flow direction** command to specify the collector flow direction.

Example

```
Switch(config-flow-record)# collect flow direction
```

Specifies to collect the direction – Ingress or Egress – of the relevant side – Initiator or Responder – of the bi-directional flow that is specified by the **initiator** keyword in the **collect connection initiator** command in the following step. Depending on the value specified by the **initiator** keyword, the **flow direction** keyword takes the following values:

- 0x01 = Ingress Flow
- 0x02 = Egress Flow

When the **initiator** keyword is set to initiator, the flow direction is specified from the initiator side of the flow. When the initiator keyword is set to responder, the flow direction is specified from the responder side of the flow. For wired AVC, the **initiator** keyword is always set to initiator.

b) Use the **collect connection initiator** command to specify the collector side of the flow (initiator or responder).

Example

```
Switch(config-flow-record)# collect connection initiator
```

Direction of the flow specified – **collect flow direction** command.

The **initiator** keyword provides the information about the direction of the flow:

- 0x01 = Initiator - the flow source is the initiator of the connection.

For wired AVC, the **initiator** keyword is always set to initiator.

7. Use these commands to collect connection initiation and traffic counters for client and server.

a) Use the **collect connection new-connections** command to specify the number of connection initiations observed.

Example

```
Switch(config-flow-record)# collect connection new-connections
```

b) Use the **collect connection client counter packets long** command to specify the number of packets sent by the client.

Example

```
Switch(config-flow-record)# collect connection client counter packets long
```

c) Use the **collect connection client counter bytes network long** command to specify the total number of bytes transmitted by the client.

Example

```
Switch(config-flow-record)# collect connection client counter bytes network
 long
```

d) Use the **collect connection server counter packets long** command to specify the number of packets sent by the server.

Example

```
Switch(config-flow-record)# collect connection server counter packets long
```

e) Use the **collect connection server counter bytes network long** command to specify the the total number of bytes transmitted by the server.

Example

```
Switch(config-flow-record)# collect connection server counter bytes network
 long
```

8. Use these command to collect absolute timestamps for the first and last packets in the flow.

a) Use the **collect timestamp absolute first** command to specify the time, in milliseconds, when the first packet was seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute first
```

b) Use the **collect timestamp absolute last** command to specify the time, in milliseconds, when the most recent packet was seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute last
```

9. Use these commands to Exit configuration mode and verify the flow record.

a) Use the **end** command to return to the privileged EXEC mode.

Example

```
Switch(config-if)# end
```

b) Use the **show flow record** command to display information about all the flow records.

Example

```
Switch# show flow record
```

Configure a flow record for ingress traffic

Learn how to configure a flow record for ingress traffic on Cisco switches. Use this procedure to enter configuration mode, define match criteria, collect statistics, and verify the flow record.

Perform these steps to create an ingress directional flow record:

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **flow record** *flow_record_name* command to enter flow record configuration mode

   Example

   ```
   Switch(config)# flow record fr-wdavc-1
   ```

3. (Optional) Use the **description** *description* command to create a description for the flow record.

   Example

   ```
   Switch(config-flow-record)# description fr-wdavc-1
   ```

4. Use these command to configure match criteria for IP version, protocol, and application name.
   a) Use the **match ipv4 version** command to specify a match to the IP version from the IPv4 header.

   Example

   ```
   Switch(config-flow-record)# match ipv4 version
   ```

   b) Use the **match ipv4 protocol** command to specify a match to the IPv4 protocol.

   Example

   ```
   Switch(config-flow-record)# match ipv4 protocol
   ```

c) Use the **match ipv4 source address** command to specify a match to the IPv4 source address as a key field.

Example

```
Switch(config-flow-record)# match ipv4 source address
```

d) Use the **match ipv4 destination address** command to specify a match to the IPv4 destination address as a key field.

Example

```
Switch(config-flow-record)# match ipv4 destination address
```

e) Use the **match transport source-port** command to specify a match to the transport source port as a key field.

Example

```
Switch(config-flow-record)# match transport source-port
```

f) Use the **match transport destination-port** command to specify a match to the transport destination port as a key field.

Example

```
Switch(config-flow-record)# match transport destination-port
```

g) Use the **match interface input** command to specify a match to the input interface as a key field.

Example

```
Switch(config-flow-record)# match interface input
```

h) Use the **match application name** command to specify a match to the application name.

Example

```
Switch(config-flow-record)# match application name
```

Specifies a match to the application name.

> **Note**
>
> This action is mandatory for AVC support, as this allows the flow to be matched against the application.

5. Use these commands to collect statistics.

a) Use the **collect interface output** command to specify the output interface from the flows.

Example

```
Switch(config-flow-record)# collect interface output
```

b) Use the **collect counter bytes long** command to specify the number of bytes in a flow.

Example

```
Switch(config-flow-record)# collect counter bytes long
```

c) Use the **collect counter packets long** command to specify the number of packets in a flow.

Example

```
Switch(config-flow-record)# collect counter packets long
```

d) Use the **collect timestamp absolute first** command to specify the time, in milliseconds when the first packet is seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute first
```

e) Use the **collect timestamp absolute last** command to specify the time, in milliseconds when the first packet is seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute last
```

6. Use these commands to Exit configuration mode and verify the flow record.

a) Use the **end** command to return to the privileged EXEC mode.

Example

```
Switch(config-if)# end
```

b) Use the **show flow record** command to display information about all the flow records.

Example

```
Switch# show flow record
```

Configure a flow record for egress traffic

Learn how to configure a flow record for egress traffic on Cisco switches. Use this procedure to create an egress directional flow record and collect statistics for monitoring and analysis.

Perform these steps to create an egress directional flow record.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

Example

```
Switch# configure terminal
```

2. Use the **flow record** *flow_record_name* command to enter flow record configuration mode

   Example

   ```
   Switch(config)# flow record fr-wdavc-1
   ```

3. (Optional) Use the **description** *description* command to create a description for the flow record.

   Example

   ```
   Switch(config-flow-record)# description fr-wdavc-1
   ```

4. Use these command to configure match criteria for IP version, protocol, and application name.
   a) Use the **match ipv4 version** command to specify a match to the IP version from the IPv4 header.

      Example

      ```
      Switch(config-flow-record)# match ipv4 version
      ```

   b) Use the **match ipv4 protocol** command to specify a match to the IPv4 protocol.

      Example

      ```
      Switch(config-flow-record)# match ipv4 protocol
      ```

   c) Use the **match ipv4 source address** command to specify a match to the IPv4 source address as a key field.

      Example

      ```
      Switch(config-flow-record)# match ipv4 source address
      ```

   d) Use the **match ipv4 destination address** command to specify a match to the IPv4 destination address as a key field.

      Example

      ```
      Switch(config-flow-record)# match ipv4 destination address
      ```

   e) Use the **match transport source-port** command to specify a match to the transport source port as a key field.

      Example

      ```
      Switch(config-flow-record)# match transport source-port
      ```

   f) Use the **match transport destination-port** command to specify a match to the transport destination port as a key field.

Example

```
Switch(config-flow-record)# match transport destination-port
```

g) Use the **match interface output** command to specify a match to the input interface as a key field.

Example

```
Switch(config-flow-record)# match interface output
```

h) Use the **match application name** command to specify a match to the application name.

Example

```
Switch(config-flow-record)# match application name
```

Specifies a match to the application name.

> **Note**
>
> This action is mandatory for AVC support, as this allows the flow to be matched against the application.

5. Use these commands to collect statistics.

a) Use the **collect interface input** command to specify the input interface from the flows.

Example

```
Switch(config-flow-record)# collect interface input
```

b) Use the **collect counter bytes long** command to specify the number of bytes in a flow.

Example

```
Switch(config-flow-record)# collect counter bytes long
```

c) Use the **collect counter packets long** command to specify the number of packets in a flow.

Example

```
Switch(config-flow-record)#
                                collect counter packets long
```

d) Use the **collect timestamp absolute first** command to specify the time, in milliseconds when the first packet is seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute first
```

e) Use the **collect timestamp absolute last** command to specify the time, in milliseconds when the first packet is seen in the flow.

Example

```
Switch(config-flow-record)# collect timestamp absolute last
```

6. Use these commands to Exit configuration mode and verify the flow record.
   a) Use the **end** command to return to the privileged EXEC mode.

Example

```
Switch(config-if)# end
```

   b) Use the **show flow record** command to display information about all the flow records.

Example

```
Switch# show flow record
```

Configure a flow monitor

Learn how to configure a flow monitor by creating and associating it with a flow record, and monitor or clear flow statistics on your device.

You can create a flow monitor and associate it with a flow record.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

Example

```
Switch# configure terminal
```

2. Use the **flow monitor** monitor-name command to create a flow monitor and enters flow monitor configuration mode.

Example

```
Switch(config)# flow monitor flow-monitor-1
```

3. (Optional) Use the **description** *description* command to create a description for the flow record.

Example

```
Switch(config-flow-monitor)# description flow-monitor-1
```

4. Use the **record** *record-name* command to specify the name of a record that was created previously.

Example

```
Switch(config-flow-monitor)# record flow-record-1
```

5. Use the **exporter** *exporter-name* command to specify the name of an exporter that was created previously.

   Example

   ```
   Switch(config-flow-monitor)# exporter flow-exporter-1
   ```

6. (Optional) Use the **cache entries** *number-of-entries* **timeout active inactive type normal** command to configure flow cache parameters.

   Example

   ```
   Switch(config-flow-monitor)# cache timeout active 1800
   ```

   Example

   ```
   Switch(config-flow-monitor)# cache timeout inactive 200
   ```

   Example

   ```
   Switch(config-flow-monitor)# cache type normal
   ```

   entries *number-of-entries* – Specifies the maximum number of flow entries in the flow cache in the range from 16 to 65536.

   > **Note**
   >
   > Only normal cache type is supported.

7. Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

8. Use these commands to monitor flow monitor statistics.
   a) Use the **show flow monitor** command to displays information about all the flow monitors.

      Example

      ```
      Switch# show flow monitor
      ```

   b) Use the **show flow monitor** *flow-monitor-name* command to display information about the specified wired AVC flow monitor.

Example

```
Switch# show flow monitor flow-monitor-1
```

c) Use the **show flow monitor** *flow-monitor-name* **statistics** command to display statistics for wired AVC flow monitor.

Example

```
Switch# show flow monitor flow-monitor-1 statistics
```

d) Use the **show flow monitor** *flow-monitor-name* **cache format table** command to display flow cache contents in a tabular format.

Example

```
Switch# show flow monitor flow-monitor-1 cache format table
```

e) Use the **show flow monitor** *flow-monitor-name* **cache format record** command to display flow cache contents in similar format as the flow record.

Example

```
Switch# show flow monitor flow-monitor-1 cache format record
```

f) Use the **show flow monitor** *flow-monitor-name* **cache format csv** command to display flow cache contents in CSV format.

Example

```
Switch# show flow monitor flow-monitor-1 cache format csv
```

9. Use the **clear flow monitor** *flow-monitor-name* **statistics** command to clear the statistics of the specified flow monitor.

Example

```
Switch# clear flow monitor flow-monitor-1 statistics
```

Clears the statistics of the specified flow monitor. Use the **show flow monitor flow-monitor-1 statistics** command after using the **clear flow monitor flow-monitor-1 statistics** to verify that all the statistics have been reset.

Associate flow monitor to an interface

Learn how to associate a flow monitor to an interface using wired AVC monitors and predefined records. Use this procedure to configure input and output packet monitoring on a Cisco switch interface.

You can attach two different wired AVC monitors with different predefined records to an interface at the same time.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **interface** *interface-id* command to specify interface and enter interface configuration mode.

   Example

   ```
   Switch(config)# interface Gigabitethernet 1/1
   ```

3. Use the **ip flow monitor** *monitor-name* **input output** command to associate a flow monitor to the interface for input and/or output packets.

   Example

   ```
   Switch(config-if) # ip flow monitor flow-monitor-1 input
   ```

4. Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

NBAR2 custom applications

This topic explains how NBAR2 supports the identification and customization of local and specific applications using custom protocols, including the types and methods of customization available.

NBAR2 custom applications are user-defined protocols that allow identification of local or specific applications not covered by the default NBAR2 protocol pack. The custom applications

- enable recognition of organization- or geography-specific applications, and
- take precedence over built-in protocols and can include selector IDs for reporting.

The custom applications allow customization using the **ip nbar custom** *myappname* command in the global configuration mode.

NBAR2 custom application customization types and methods

NBAR2 supports these types of application customization.

- Generic protocol customization: HTTP, SSL, DNS
- Composite customization: Based on multiple underlying protocols, such as **server-name** .
- Layer3 or Layer4 customization: IPv4 address, DSCP values, TCP or UDP ports, flow source or destination direction
- Byte Offset customization: Based on specific byte values in the payload. the supported customizations are:

- HTTP
- SSL
- DNS
- Composite
- L3/L4

HTTP customization

This topic explains how HTTP customization enables the identification and classification of network traffic based on various HTTP fields, allowing for tailored application recognition and control.

HTTP customization is a method that allows traffic identification based on these HTTP fields

- cookie - HTTP Cookie
- host - Host name of Origin Server containing resource
- method - HTTP method
- referrer - Address the resource request was obtained from
- url - Uniform Resource Locator path
- user-agent - Software used by agent sending the request
- version - HTTP version
- via - HTTP via field

Define custom applications that match specific HTTP field values to implement HTTP customization.

You can create a custom application named MYHTTP that matches the HTTP host "*mydomain.com" with selector ID 10 by entering the `ip nbar custom MYHTTP http host *mydomain.com id 10` command in configuration mode.

SSL customization

This topic explains how SSL customization enables the handling of SSL encrypted traffic by using information extracted from the SSL Server Name Indication (SNI) or Common Name (CN). It describes the process and provides an example configuration.

SSL customization is a feature that allows you to manage SSL encrypted traffic by extracting information from the SSL Server Name Indication (SNI) or Common Name (CN).

You can create a custom application named MYSSL that matches the SSL unique-name"*mydomain.com" with selector ID 11 by entering the `ip nbar custom MYSSL ssl unique-name *mydomain.com id 11` command in configuration mode.

DNS customization

This topic explains how DNS customization enables NBAR2 to examine DNS traffic, correlate DNS responses to applications, and use cached IP addresses for subsequent packet flows. It also covers the commands used for DNS-based application customization.

DNS customization is a feature that allows NBAR2 to analyze DNS request and response traffic, correlate DNS responses to applications, and cache returned IP addresses used for later packet flows associated with that specific application.

Define and extend applications based on DNS domains

DNS customization enables the use of specific commands to define or extend applications based on DNS domain names.

Use the **ip nbar custom***application-name***dns***domain-name* **id***application-id* command for DNS customization.

Use the **ip nbar custom***application-name***dns domain-name** *domain-name***extends***existing-application* command to extend an existing application.

For more information on DNS-based customization, see http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/configuration/xe-3s/asr1000/qos-nbar-xe-3s-asr-1000-book/nbar-custapp-dns-xe.html.

You can create a custom application named MYDNS that matches the DNS domain name "*mydomain.com" with selector ID 12 by entering the **`ip nbar custom MYDNS dns domain-name *mydomain.com id 12`** command in configuration mode.

Composite customization

This topic explains how NBAR2 provides a way to customize applications based on domain names appearing in HTTP, SSL, or DNS. Composite Customization enables the creation of custom applications using domain-based selectors.

Composite customization is a feature in NBAR2 that allows users to define custom applications by specifying domain names found in HTTP, SSL, or DNS traffic.

You can create a custom application named MYDOMAIN using HTTP, SSL, or DNS domain name "*mydomain.com" with selector ID 13 by entering the **`ip nbar custom MYDOMAIN composite server-name *mydomain.com id 13`** command in configuration mode.

L3/L4 Customization

This topic explains how Layer3/Layer4 customization is based on the packet tuple and is always matched on the first packet of a flow.

Layer3 or layer4 customization is a process that matches traffic based on the packet tuple and always applies to the first packet of a flow.

You can create a custom application named LAYER4CUSTOM matching IP addresses 10.56.1.10 and 10.56.1.11, TCP and DSCP ef with selector ID 14 by entering the **`ip nbar custom LAYER4CUSTOM transport tcp id 14`** command in configuration mode. Here is an example.

```
Switch# configure terminal
Switch(config)# ip nbar custom LAYER4CUSTOM transport tcp id 14
Switch(config-custom)# ip address 10.56.1.10 10.56.1.11
Switch(config-custom)# dscp ef
```

Monitor custom applications

This topic explains how to monitor custom applications using show commands on Cisco devices, including protocol identification and discovery for custom application types.

This topic describes the use of show commands to monitor custom applications and protocols on Cisco devices.

Show commands to monitor custom applications

Use the **show ip nbar protocol-id | inc Custom** command to monitor nbar custom statistics.

```
Device# show ip nbar  protocol-id | inc Custom
LAYER4CUSTOM              14            Custom
MYDNS                     12            Custom
MYDOMAIN                  13            Custom
```

```
MYHTTP                    10            Custom
MYSSL                     11            Custom
```

Use the **show ip nbar protocol-discovery protocol** *CUSTOM_APP* command to monitor nbar custom specific (here it is SSL) statistics.

```
Switch# show ip nbar  protocol-id MYSSL
Protocol Name              id             type
-----------------------------------------------
MYSSL                     11            Custom
```

NBAR2 dynamic hitless protocol pack

This topic explains the NBAR2 dynamic hitless protocol pack, which updates NBAR2 protocol support on a device without replacing the Cisco software, and describes its features and usage.

NBAR2 dynamic hitless protocol packs are software packages that update protocol support on your device without upgrading the Cisco software. Each pack includes application signatures and attributes. Every software release contains a built-in protocol pack.

- Protocol packs are easy and fast to load.
- They are easy to upgrade to a higher version or revert to a lower version protocol pack.
- They do not require the switch to be reloaded.

Update NBAR2 protocol pack

If a new application protocol needs to be recognized by NBAR2, a protocol pack update can be loaded without reloading the switch or replacing the Cisco software.

> **Note**
>
> Download the latest protocol packs from the Cisco Software center at https://software.cisco.com/download/home.

Load NBAR2 protocol pack

Learn how to load the NBAR2 protocol pack on your Switch to enable advanced protocol recognition and management.

Perform these steps to load the NBAR2 protocol pack.

Before you begin

Before loading a new protocol pack, you must copy the protocol pack to the flash on all the switch members.

Procedure

1. Use the **enable** command to enter the privileged EXEC mode.

Example

```
Switch> enable
```

Enter your password if prompted.

2. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

3. Use the **ip nbar protocol-pack** *protocol-pack* [**force**] command to load the protocol pack.

   Example

   ```
   Switch(config)# ip nbar protocol-pack flash:defProtoPack
   ```

   Use the **force** keyword to specify and load a protocol pack of a lower version, which is different from the base protocol pack version. Doing so also removes the configuration that is not supported by the current protocol pack on the switch.

   Example

   ```
   Switch(config)# ip nbar protocol-pack flash:OldDefProtoPack force
   ```

   Use the **default ip nbar protocol-pack** command to revert to the built-in protocol pack.

   Example

   ```
   Switch(config)# default ip nbar protocol-pack
   ```

4. Use the **end** command to return to privileged EXEC mode.

   Example

   ```
   Switch(config-if)# end
   ```

5. (Optional) Use the **show ip nbar protocol-pack** {protocol-pack | **active** } [**detail**] command to display the protocol pack information.

   Example

   ```
   Switch# show ip nbar protocol-pack active
   ```

   - Verify the loaded protocol pack version, publisher, and other details using this command.
   - Use the *protocol-pack* argument to display information about the specified protocol pack.
   - Use the **active** keyword to display active protocol pack information.
   - Use the **detail** keyword to display detailed protocol pack information.

Monitor AVC statistics on the switch

Learn how to monitor application visibility and statistics on the switch and access ports using AVC-related commands.

Use this procedure to monitor application visibility and statistics on the switch and access ports.

These commands can be used to monitor application visibility on the switch and access ports.

Procedure

1. (Optional) Use the **show ip nbar protocol-discovery interface** *interface-type interface-number* **stats byte-count bit-rate packet-count max-bit-rate protocol** *protocol-name* **top-n** *number* command to monitor the statistics for all interfaces on which the NBAR Protocol Discovery feature is enabled are displayed.

   Table 1: Syntax description

Command Syntax	Description
interface	(Optional) Specifies that Protocol Discovery statistics for the interface are to be displayed.
type	Type of interface or subinterface whose policy configuration is to be displayed.
number	Port, connector, VLAN, or interface card number.
stats	(Optional) Specifies that the byte count, byte rate, or packet count is to be displayed.
byte-count	(Optional) Specifies that the byte count is to be displayed.
max-bit-rate	(Optional) Specifies that the maximum bit rate is to be displayed.
packet-count	(Optional) Specifies that the packet count is to be displayed.
protocol	(Optional) Specifies that statistics for a specific protocol are to be displayed.
protocol-name	(Optional) User-specified protocol name for which the statistics are to be displayed.
top-n	(Optional) Specifies that a top-n is to be displayed. A top-n is the number of most active NBAR-supported protocols, where n is the number of protocols to be displayed. For instance, if top-n 3 is entered, the three most active NBAR-supported protocols will be displayed.
number	(Optional) Specifies the number of most active NBAR-supported protocols to be displayed.

2. (Optional) Use the **show policy-map interface** *interface-type interface-number* command to display information of policy map applied to the interface. For details com configuration, refer to AVC configuration examples on page 34

AVC configuration examples

This topic explains how to configure AVC class maps, policy maps, and monitor configurations using show commands on Cisco switches. Examples and sample outputs are provided for both basic and attribute-based QoS.

This topic provides examples for creating class maps, policy maps, and monitoring configurations for AVC on Cisco switches.

This example shows how to create class maps with apply match protocol filters for application name:

```
Switch# configure terminal
Switch(config)# class-map match-any NBAR-VOICE
Switch(config-cmap)# match protocol ms-lync-audio
Switch(config-cmap)# end
```

This example shows how to create policy maps and define existing class maps for egress QoS.

```
Switch # configure terminal
Switch(config)# policy-map test-avc-up
Switch(config-pmap)# class cat-browsing
Switch(config-pmap-c)# police 150000
Switch(config-pmap-c)# set dscp 12
Switch(config-pmap-c)# end
```

This example shows how to create policy maps and define existing class maps for ingress QoS.

```
Switch# configure terminal
Switch(config)# policy-map test-avc-down
Switch(config-pmap)# class cat-browsing
Switch(config-pmap-c)# police 200000
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# end
```

This example shows how to apply policy maps to a switch port.

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet 1/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 20
Switch(config-if)# service-policy input POLICING_IN
Switch(config-if)# end
```

This example shows how to create class maps based on NBAR attributes.

```
Switch# configure terminal
Switch(config)# class-map match-all rel-relevant
Switch(config-cmap)# match protocol attribute business-relevance
business-relevant
Switch(config)# class-map match-all rel-irrelevant
Switch(config-cmap)# match protocol attribute business-relevance
business-irrelevant
Switch(config)# class-map match-all rel-default
Switch(config-cmap)# match protocol attribute business-relevance default
Switch(config)# class-map match-all class--ops-admin-and-rel
Switch(config-cmap)# match protocol attribute traffic-class ops-admin-mgmt
Switch(config-cmap)# match protocol attribute business-relevance
business-relevant
```

This example shows how to create policy maps based on class maps based on NBAR attributes.

```
Switch# configure terminal
Switch(config)# policy-map attrib--rel-types
Switch(config-pmap)# class rel-relevant
Switch(config-pmap-c)# set dscp ef
```

```
Switch(config-pmap-c)# class rel-irrelevant
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap-c)# class rel-default
Switch(config-pmap-c)# set dscp default
Switch(config)# policy-map attrib--ops-admin-and-rel
Switch(config-pmap)# class class--ops-admin-and-rel
Switch(config-pmap-c)# set dscp cs5
```

This example shows how to attach a policy map based on NBAR attributes to a wired port.

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/2
Switch(config-if)# service-policy input attrib--rel-types
```

Show commands to monitor the configuration

Use the **show ip nbar protocol-discovery** command to display a report of the protocol discovery statistics per interface.

This is a sample output for the statistics per interface.

```
Switch# show ip nbar protocol-discovery interface GigabitEthernet1/1
GigabitEthernet1/1
Last clearing of "show ip nbar protocol-discovery" counters 00:03:16
                                Input                                Output
Protocol                        Packet Count                         Packet
Count
                                Byte Count                           Byte
Count
                                30sec Bit Rate (bps)                 30sec
Bit Rate (bps)
                                30sec Max Bit Rate (bps)             30sec
Max Bit Rate (bps)
------------------------ ------------------------
-----------------------------------------------------
ms-lync                         60580                                55911
                               31174777                             28774864

                                3613000                              93000
                                3613000                              3437000
Total                           60580                                55911
                               31174777                             28774864

                                3613000                              93000
                                3613000                              3437000

                               -------                              --------
```

Use the **show policy-map interface** command to display the QoS statistics and the configured policy maps on all interfaces.

This is a sample output for the policy-maps configured on all the interfaces:

```
Switch# show policy-map int
GigabitEthernet1/1
 Service-policy input: MARKING-IN
  Class-map: NBAR-VOICE (match-any)
    718 packets
    Match: protocol ms-lync-audio
      0 packets, 0 bytes
```

```
      30 second rate 0 bps
   QoS Set
      dscp ef
   Class-map: NBAR-MM_CONFERENCING (match-any)
      6451 packets
      Match: protocol ms-lync
        0 packets, 0 bytes
        30 second rate 0 bps
      Match: protocol ms-lync-video
        0 packets, 0 bytes
        30 second rate 0 bps
      QoS Set
        dscp af41
    Class-map: class-default (match-any)
      34 packets
      Match: any
```

Show commands to monitor attributes-based QoS configuration

Use the **show policy-map interface** command to display the attribute-based QoS statistics and the configured policy maps on all interfaces.

This sample output for the policy-maps configured on all the interfaces:

```
Switch# show policy-map interface gigabitEthernet 1/2
GigabitEthernet1/2
  Service-policy input: attrib--rel-types
    Class-map: rel-relevant (match-all)
      20 packets
      Match: protocol attribute business-relevance business-relevant
      QoS Set
        dscp ef
    Class-map: rel-irrelevant (match-all)
      0 packets
      Match: protocol attribute business-relevance business-irrelevant
      QoS Set
        dscp af11
    Class-map: rel-default (match-all)
      14 packets
      Match: protocol attribute business-relevance default
      QoS Set
        dscp default
    Class-map: class-default (match-any)
      0 packets
      Match: any
```

Use the **show ip nbar protocol-attribute** command to display all the protocol attributes used by NBAR.

This example shows sample output for some of the attributes.

```
Switch# show ip nbar protocol-attribute cisco-jabber-im
 Protocol Name : cisco-jabber-im
              encrypted : encrypted-yes
                 tunnel : tunnel-no
               category : voice-and-video
         sub-category : enterprise-media-conferencing
      application-group : cisco-jabber-group
         p2p-technology : p2p-tech-no
          traffic-class : transactional-data
```

```
      business-relevance : business-relevant
         application-set : collaboration-apps
```

```
Switch# show ip nbar protocol-attribute google-services
   Protocol Name : google-services
              encrypted : encrypted-yes
                 tunnel : tunnel-no
               category : other
           sub-category : other
       application-group : google-group
          p2p-technology : p2p-tech-yes
           traffic-class : transactional-data
      business-relevance : default
         application-set : general-browsing
```

```
Switch# show ip nbar protocol-attribute dns
  Protocol Name : google-services
              encrypted : encrypted-yes
                 tunnel : tunnel-no
               category : other
           sub-category : other
       application-group : google-group
          p2p-technology : p2p-tech-yes
           traffic-class : transactional-data
      business-relevance : default
         application-set : general-browsing
```

```
Switch# show ip nbar protocol-attribute unknown
      Protocol Name : unknown
              encrypted : encrypted-no
                 tunnel : tunnel-no
               category : other
           sub-category : other
       application-group : other
          p2p-technology : p2p-tech-no
           traffic-class : bulk-data
      business-relevance : default
         application-set : general-misc
```

Show commands to monitor flow monitor configuration

Use the **show flow monitor wdavc** command to display information about the specified wired AVC flow monitor.

```
Switch# show flow monitor wdavc
Flow Monitor wdavc:
  Description:       User defined
  Flow Record:       wdavc
  Flow Exporter:     wdavc-exp (inactive)
  Cache:
    Type:                 normal (Platform cache)
    Status:               not allocated
    Size:                 12000 entries
    Inactive Timeout:     15 secs
    Active Timeout:       1800 secs
```

Use the **show flow monitor wdavc statistics** command to monitor for wired AVC flow monitor statistics.

```
Switch# show flow monitor wdavc statistics
Cache type:                                 Normal (Platform cache)
  Cache size:                                  12000
  Current entries:                                13
  Flows added:                                    26
  Flows aged:                                     13
    - Active timeout      (  1800 secs)           1
    - Inactive timeout    (    15 secs)          12
```

Use the **clear flow monitor wdavc statistics** command to clear the specified flow monitor. statistics.

Use the **show flow monitor wdavc statistics** command after using the **clear flow monitor wdavc statistics** to verify that all the statistics have been reset. The following is a sample output of the **show flow monitor wdavc statistics** command after clearing flow monitor statistics.

```
Switch# show flow monitor wdavc statistics
Cache type: Normal (Platform cache)
  Cache size: 12000
  Current entries: 0
  Flows added: 0
  Flows aged: 0
```

Show commands to monitor cache information

Use the **show flow monitor wdavc cache format table** command to display flow cache contents in a tabular format.

```
Switch# show flow monitor wdavc cache format table
Cache type:                                 Normal (Platform cache)
  Cache size:                                  12000
  Current entries:                                13
  Flows added:                                    26
  Flows aged:                                     13
    - Active timeout      (  1800 secs)           1
    - Inactive timeout    (    15 secs)          12
CONN IPV4 INITIATOR ADDR  CONN IPV4 RESPONDER ADDR  CONN RESPONDER PORT  FLOW
 OBSPOINT ID  IP VERSION  IP PROT  APP NAME                      flow dirn
.....................
------------------------  ------------------------  -------------------
----------------  ----------  -------  ----------------------------  ---------

64.103.125.147            144.254.71.184                              53
  4294967305           4       17  port dns                     Input
.....................
64.103.121.103            10.1.1.2                                    67
  4294967305           4       17  layer7 dhcp                  Input
....contd............
64.103.125.3              64.103.125.97                               68
  4294967305           4       17  layer7 dhcp                  Input
.....................
10.0.2.6                  157.55.40.149                              443
  4294967305           4        6  layer7 ms-lync               Input
.....................
64.103.126.28             66.163.36.139                              443
  4294967305           4        6  layer7 cisco-jabber-im       Input
....contd............
64.103.125.2              64.103.125.29                               68
  4294967305           4       17  layer7 dhcp                  Input
.....................
64.103.125.97             64.103.101.181                              67
```

```
  4294967305           4       17  layer7 dhcp                   Input
.....................
192.168.100.6             10.10.20.1                                5060
  4294967305           4       17  layer7 cisco-jabber-control   Input
....contd...........
64.103.125.3              64.103.125.29                             68
  4294967305           4       17  layer7 dhcp                   Input
.....................
10.80.101.18              10.80.101.6                               5060
  4294967305           4        6  layer7 cisco-collab-control   Input
.....................
10.1.11.4                 66.102.11.99                              80
  4294967305           4        6  layer7 google-services        Input
....contd...........
64.103.125.2              64.103.125.97                             68
  4294967305           4       17  layer7 dhcp                   Input
.....................
64.103.125.29             64.103.101.181                            67
  4294967305           4       17  layer7 dhcp                   Input
.....................
```

Use the **show flow monitor wdavc cache format record** command to display flow cache contents in similar format as the flow record.

```
Switch# show flow monitor wdavc cache format record
Cache type:                               Normal (Platform cache)
  Cache size:                                12000
  Current entries:                              13
  Flows added:                                  26
  Flows aged:                                   13
    - Active timeout      (  1800 secs)          1
    - Inactive timeout    (    15 secs)         12
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.147
CONNECTION IPV4 RESPONDER ADDRESS:        144.254.71.184
CONNECTION RESPONDER PORT:                53
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         port dns
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:46.917
connection initiator:                     Initiator
connection count new:                     2
connection server packets counter:        1
connection client packets counter:        1
connection server network bytes counter:  190
connection client network bytes counter:  106
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.121.103
CONNECTION IPV4 RESPONDER ADDRESS:        10.1.1.2
CONNECTION RESPONDER PORT:                67
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 dhcp
flow direction:                           Input
timestamp abs first:                      08:55:47.917
timestamp abs last:                       08:55:47.917
connection initiator:                     Initiator
connection count new:                     1
connection server packets counter:        0
connection client packets counter:        1
```

```
connection server network bytes counter:  0
connection client network bytes counter:  350
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS:        64.103.125.97
CONNECTION RESPONDER PORT:                68
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 dhcp
flow direction:                           Input
timestamp abs first:                      08:55:47.917
timestamp abs last:                       08:55:53.917
connection initiator:                     Initiator
connection count new:                     1
connection server packets counter:        0
connection client packets counter:        4
connection server network bytes counter:  0
connection client network bytes counter:  1412
CONNECTION IPV4 INITIATOR ADDRESS:        10.0.2.6
CONNECTION IPV4 RESPONDER ADDRESS:        157.55.40.149
CONNECTION RESPONDER PORT:                443
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              6
APPLICATION NAME:                         layer7 ms-lync
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:46.917
connection initiator:                     Initiator
connection count new:                     2
connection server packets counter:        10
connection client packets counter:        14
connection server network bytes counter:  6490
connection client network bytes counter:  1639
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.126.28
CONNECTION IPV4 RESPONDER ADDRESS:        66.163.36.139
CONNECTION RESPONDER PORT:                443
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              6
APPLICATION NAME:                         layer7 cisco-jabber-im
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:46.917
connection initiator:                     Initiator
connection count new:                     2
connection server packets counter:        12
connection client packets counter:        10
connection server network bytes counter:  5871
connection client network bytes counter:  2088
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS:        64.103.125.29
CONNECTION RESPONDER PORT:                68
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 dhcp
flow direction:                           Input
timestamp abs first:                      08:55:47.917
timestamp abs last:                       08:55:47.917
connection initiator:                     Initiator
connection count new:                     1
```

```
connection server packets counter:        0
connection client packets counter:        2
connection server network bytes counter:  0
connection client network bytes counter:  712
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.97
CONNECTION IPV4 RESPONDER ADDRESS:        64.103.101.181
CONNECTION RESPONDER PORT:                67
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 dhcp
flow direction:                           Input
timestamp abs first:                      08:55:47.917
timestamp abs last:                       08:55:47.917
connection initiator:                     Initiator
connection count new:                     1
connection server packets counter:        0
connection client packets counter:        1
connection server network bytes counter:  0
connection client network bytes counter:  350
CONNECTION IPV4 INITIATOR ADDRESS:        192.168.100.6
CONNECTION IPV4 RESPONDER ADDRESS:        10.10.20.1
CONNECTION RESPONDER PORT:                5060
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 cisco-jabber-control
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:46.917
connection initiator:                     Initiator
connection count new:                     1
connection server packets counter:        0
connection client packets counter:        2
connection server network bytes counter:  0
connection client network bytes counter:  2046
CONNECTION IPV4 INITIATOR ADDRESS:        64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS:        64.103.125.29
CONNECTION RESPONDER PORT:                68
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              17
APPLICATION NAME:                         layer7 dhcp
flow direction:                           Input
timestamp abs first:                      08:55:47.917
timestamp abs last:                       08:55:47.917
connection initiator:                     Initiator
connection count new:                     1
connection server packets counter:        0
connection client packets counter:        2
connection server network bytes counter:  0
connection client network bytes counter:  712
CONNECTION IPV4 INITIATOR ADDRESS:        10.80.101.18
CONNECTION IPV4 RESPONDER ADDRESS:        10.80.101.6
CONNECTION RESPONDER PORT:                5060
FLOW OBSPOINT ID:                         4294967305
IP VERSION:                               4
IP PROTOCOL:                              6
APPLICATION NAME:                         layer7 cisco-collab-control
flow direction:                           Input
timestamp abs first:                      08:55:46.917
timestamp abs last:                       08:55:47.917
```

```
connection initiator:                       Initiator
connection count new:                       2
connection server packets counter:          23
connection client packets counter:          27
connection server network bytes counter:    12752
connection client network bytes counter:    8773
CONNECTION IPV4 INITIATOR ADDRESS:          10.1.11.4
CONNECTION IPV4 RESPONDER ADDRESS:          66.102.11.99
CONNECTION RESPONDER PORT:                  80
FLOW OBSPOINT ID:                           4294967305
IP VERSION:                                 4
IP PROTOCOL:                                6
APPLICATION NAME:                           layer7 google-services
flow direction:                             Input
timestamp abs first:                        08:55:46.917
timestamp abs last:                         08:55:46.917
connection initiator:                       Initiator
connection count new:                       2
connection server packets counter:          3
connection client packets counter:          5
connection server network bytes counter:    1733
connection client network bytes counter:    663
CONNECTION IPV4 INITIATOR ADDRESS:          64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS:          64.103.125.97
CONNECTION RESPONDER PORT:                  68
FLOW OBSPOINT ID:                           4294967305
IP VERSION:                                 4
IP PROTOCOL:                                17
APPLICATION NAME:                           layer7 dhcp
flow direction:                             Input
timestamp abs first:                        08:55:47.917
timestamp abs last:                         08:55:53.917
connection initiator:                       Initiator
connection count new:                       1
connection server packets counter:          0
connection client packets counter:          4
connection server network bytes counter:    0
connection client network bytes counter:    1412
CONNECTION IPV4 INITIATOR ADDRESS:          64.103.125.29
CONNECTION IPV4 RESPONDER ADDRESS:          64.103.101.181
CONNECTION RESPONDER PORT:                  67
FLOW OBSPOINT ID:                           4294967305
IP VERSION:                                 4
IP PROTOCOL:                                17
APPLICATION NAME:                           layer7 dhcp
flow direction:                             Input
timestamp abs first:                        08:55:47.917
timestamp abs last:                         08:55:47.917
connection initiator:                       Initiator
connection count new:                       1
connection server packets counter:          0
connection client packets counter:          1
connection server network bytes counter:    0
connection client network bytes counter:    350
```

Use the **show flow monitor wdavc cache format csv** command to display flow cache contents in CSV format.

```
Switch# show flow monitor wdavc cache format csv
Cache type:                                 Normal (Platform cache)
  Cache size:                                  12000
  Current entries:                                13
```

```
  Flows added:                                  26
  Flows aged:                                   13
    - Active timeout      (  1800 secs)          1
    - Inactive timeout    (    15 secs)         12
CONN IPV4 INITIATOR ADDR,CONN IPV4 RESPONDER ADDR,CONN RESPONDER PORT,FLOW
OBSPOINT ID,IP VERSION,IP
PROT,APP NAME,flow dirn,time abs first,time abs last,conn initiator,conn count
 new,conn server packets
cnt,conn client packets cnt,conn server network bytes cnt,conn client network
 bytes cnt
64.103.125.147,144.254.71.184,53,4294967305,4,17,port
dns,Input,08:55:46.917,08:55:46.917,Initiator,2,1,1,190,106
64.103.121.103,10.1.1.2,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
64.103.125.3,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
10.0.2.6,157.55.40.149,443,4294967305,4,6,layer7 ms-
lync,Input,08:55:46.917,08:55:46.917,Initiator,2,10,14,6490,1639
64.103.126.28,66.163.36.139,443,4294967305,4,6,layer7 cisco-jabber-
im,Input,08:55:46.917,08:55:46.917,Initiator,2,12,10,5871,2088
64.103.125.2,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
64.103.125.97,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
192.168.100.6,10.10.20.1,5060,4294967305,4,17,layer7 cisco-jabber-
control,Input,08:55:46.917,08:55:46.917,Initiator,1,0,2,0,2046
64.103.125.3,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
10.80.101.18,10.80.101.6,5060,4294967305,4,6,layer7 cisco-collab-
control,Input,08:55:46.917,08:55:47.917,Initiator,2,23,27,12752,8773
10.1.11.4,66.102.11.99,80,4294967305,4,6,layer7 google-
services,Input,08:55:46.917,08:55:46.917,Initiator,2,3,5,1733,663
64.103.125.2,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
64.103.125.29,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
```

AVC troubleshooting: questions and answers

This topic explains basic troubleshooting questions and answers for wired Application Visibility and Control on Cisco IE9300 switches.

- **Question:** My IPv6 traffic is not being classified.

  Answer: Currently only IPv4 traffic is supported.

- **Question:** My multicast traffic is not being classified.

  Answer: Currently only unicast traffic is supported.

- **Question:** I send ping but I don't see traffic being classified.

  Answer: Only TCP/UDP protocols are supported.

- **Question:** I see that most of my traffic is CAPWAP traffic, why?

  Answer: Make sure that you have enabled NBAR on an access port that is not connected to a wireless access port. All traffic coming from APs will be classified as capwap. Actual classification in this case happens either on the AP or WLC.

- **Question:** In protocol-discovery, I see traffic only on one side. Along with that, there is a lot of unknown traffic.

Answer: This usually indicates that NBAR sees asymmetric traffic: one side of the traffic is classified in one switch member and the other on a different member. The recommendation is to attach NBAR only on access ports where we see both sides of the traffic. If you have multiple uplinks, you can't attach NBAR on them due to this issue. Similar issue happens if you configure NBAR on an interface that is part of a port channel.

- **Question:** With protocol-discovery, I see an aggregate view of all application. How can I see traffic distribution over time?

  Answer: WebUI will give you view of traffic over time for the last 48 hours.

- **Question:** I can't configure queue-based egress policy with **match protocol** *protocol-name* command.

  Answer: Only **shape** and **set DSCP** are supported in a policy with NBAR2 based classifiers. Common practice is to set DSCP on ingress and perform shaping on egress based on DSCP.

- **Question:** Why can't I attach NBAR to an SVI?

  Answer: NBAR is only supported on physical interfaces.

  Question: I don't have NBAR2 attached to any interface but I still see that NBAR2 is activated.

  Answer: If you have any class-map with **match protocol** *protocol-name* , NBAR will be globally activated on the but no traffic will be subjected to NBAR classification. This is an expected behavior and it does not consume any resources.

- **Question:** I see some traffic under the default QOS queue. Why?

  Answer: For each new flow, it takes a few packets to classify it and install the result in the hardware. During this time, the classification would be 'un-known' and traffic will fall under the default queue.

Feature history for AVC in a wired network

This topic explains the feature history for Application Visibility and Control (AVC) in a wired network, including release information and platform support details.

Table 2: Feature History Table

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.7.x	Application Visibility and Control in a Wired Network	AVC is a critical part of Cisco's efforts to evolve its Branch and Campus solutions from being strictly packet and connection based to being application-aware and application-intelligent.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn .

2 Swap Drive

Topics:

- Swap Drive
- Guidelines for using SD cards and USB drives
- How Swap Drive works
- Swap the external drive
- Swap Drive commands

Swap Drive

This topic explains how the swap drive feature enables quick replacement of a failed switch by using an external SD card or USB drive to restore the original switch's image and configuration.

The Swap Drive feature is a disaster recovery mechanism that allows you to replace a failed switch by transferring an external secure digital (SD) card or USB drive containing a backup of the original switch to a new switch. It

- restores the same image and configuration to the new switch as the original switch, minimizing downtime.
- requires prior synchronization of the original switch with the SD card or USB drive, and
- supports SD cards formatted with DOSFS and USB 2.0 drives and does not support third-party SD or SDHC cards.

Switch recovery with Swap Drive

Starting from release Cisco IOS XE Cupertino 17.9.1, the swap drive feature is available for Cisco Catalyst IE9300 Rugged Series Switches.

Each Cisco Catalyst IE9300 Rugged Series Switch has a SD card slot and a USB-A port.

For information about the switch SD card slot and USB port, refer to the Cisco Catalyst IE9300 Rugged Series Switch Hardware Installation Guide.

Although synchronization can be done manually at any time, it is recommended to configure automatic synchronization at set times. When you request synchronization either manually or automatically, the switch checks for discrepancies between the internal flash drive, SD card or the USB drive.

> **Note**
>
> To restore settings to the new switch, you must have previously synchronized the original switch with the SD card or USB drive.

Swap Drive procedures and CLI commands

For Swap Drive instructions, refer to the Swap the External Drive.

For CLI commands, refer to the Swap Drive CLI Commands.

Guidelines for using SD cards and USB drives

Outlines best practices for using SD cards and USB drives with the switch.

- The switch uses the Disk Operating System Filing System (DOSFS) to format SD cards.
- The switch supports only Cisco-approved SD cards and excludes third-party SD cards or SD High Capacity (SDHC) cards.
- The switch reads data from a write-protected SD card during the boot process but prevents any updates to it.
- You can use a USB 2.0 drive as an alternative to an SD card for the swap drive feature.

How Swap Drive works

Describes the process by which swap drives enable migration of system images and configurations between switches, ensuring efficient replacement or deployment.

This process ensures that system images and configurations are preserved and can be easily transferred between devices, facilitating efficient switch replacement or deployment.

Summary

Swap drives enable the migration of system images and configurations between switches by utilizing external storage media. The key components involved in the process are:

- **Switch**: The source or target device for system data.
- **External storage**: An SD card or USB drive that stores the system image and configuration.

Workflow

The process involves these stages:

1. Backup: The system duplicates the entire configuration and image to the external SD card or USB drive. You can trigger this operation using a CLI command for one-time or periodic synchronization. If the switch has already been backed up, only changes made since the last backup are duplicated to the external drive.
2. Recovery: Restore occurs automatically when the external drive is inserted into the new switch and powered on. The switch scans for an SD card with image and configuration.
   - If the image and configuration are present on the SD card, the switch then copies them to internal flash and comes up with the image and configuration of the original system.
   - If an SD card is not present or does not have the original switch's image and configuration, the switch checks the USB drive.
   - If the image and configuration are present on the USB drive, the switch comes up with the original switch's image and configuration.
3. Initial configuration: If the image and configuration are not present either in SD card or USB, the system prompts the user to continue with initial configuration.

Swap the external drive

Learn how to swap the external SD card or USB drive between switches to recover configuration and image after a hardware failure.

You can remove the SD card from the failed switch, insert it into the new switch, and power on the new switch. A new switch has no startup configuration file because it has never been deployed.

Before you begin

Power off the switch, transfer the SD card or USB drive, and connect the network cables.

Procedure

1. On the failed switch, remove the SD card or USB drive.

> **Note**
>
> The SD card and USB drive are hot-swappable. However, do not remove either device from the switch while sdflash write is in progress.

2. On the new switch, ensure that the SD card or USB drive is oriented properly, and then press it into the slot on the switch until it is seated.
3. Power on the new switch.

   The image and configuration of the failed switch are transferred to the new one.

Swap Drive commands

Lists and describes Swap Drive commands for synchronizing, configuring, and checking switch images and configurations.

Use this reference to understand command functions and requirements.

> **Note**
>
> Run all **auto sync** commands in configuration mode.

Table 3: Swap Drive commands and their descriptions

Command	Description
• **sync sdflash:** • **sync usbflash1:**	Synchronizes the switch image and configuration files from internal flash to the SD or USB drive.
• **sync sdflash: ios-image** • **sync usbflash1: ios-image**	Synchronizes the switch image from the internal flash to the SD or USB drive.
• **sync sdflash:skip config** • **sync usbflash1: skip config**	Synchronizes the switch image from the internal flash to the SD or USB drive but does not sync the configuration.
• **sync sdflash:skip ios-image** • **sync usbflash1: skip ios-image**	Synchronizes the configuration files from the internal flash to the SD or USB drive but does not sync the image.
sync restore-bundle	Copies the bundle image to the new switch without installing it during the restore process.

Command	Description
[no] auto sync enable	Enables or disables the auto sync feature. **Note** Auto synchronization is disabled by default. You must enable auto sync before using other options for this feature.
auto sync config [usbflash1|sdflash]:	Specifies the configuration that runs during synchronization.
[no] auto sync run time: [hh:mm:ss]	Specifies the time at which the synchronization is performed. By default, the time is set to 00:00:00.
show sync status	Displays the last synchronization time and status. **Note** If a type-6 password is configured, the status displays the configuration as out of sync and includes a message stating that type-6 passwords are not synced.
show auto sync configuration	Displays all the configuration settings.
show auto sync status	Displays the last auto synchronization time and status.

3 Dying Gasp

Topics:

- Dying Gasp
- Enable dying-gasp notification

Dying Gasp

This topic explains the Dying Gasp feature, which provides a warning when power to a network device fails and enables a temporary power supply for graceful shutdown and notification to management systems.

The Dying Gasp feature is a network device capability that provides a warning when power fails, enabling a temporary power supply for graceful shutdown and sends a message or signal to the network management system (NMS) through an SNMP trap. Dying Gasp feature

- enables a temporary backup power supply (capacitor) lasting 5 to 10 milliseconds for a graceful shutdown and message generation, and
- sends a `dying-gasp` message or SNMP trap to the network management system (NMS) before shutting down upon power failure.

Dying Gasp packets can be sent to a maximum of five servers for each notification type.

Note

The Dying Gasp feature works only in standalone Cisco Catalyst IE9300 Rugged Series Switches.

Enable dying-gasp notification

Learn how to enable dying-gasp notification through syslog, SNMP trap, or Ethernet OAM using the dying-gasp command.

Use the **dying-gasp** command to enable dying-gasp notification through syslog, SNMP trap, or Ethernet OAM.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **dying-gasp primary {syslog | snmp-trap | ethernet-oam} secondary {syslog | snmp-trap | ethernet-oam}** command to configure dying-gasp notification.

   Table 4: Syntax description

Command syntax	Description
primary	Configures Dying-gasp primary notification.
secondary	configures Dying-gasp secondary notification.

Command syntax	Description
ethernet-oam	Enables Ethernet-OAM notification command. **Note** **ethernet-oam** is not supported on Cisco Catalyst IE9300 Rugged Series Switches .
snmp-trap	Sends trap notification to SNMP server command.
syslog	Enables system logger.

Use the **snmp-server host** command to specify the SNMP server for the dying-gasp message.

Use the **logging host hostname-or-ipaddress transport udp** command to specify the syslog server sending the syslog dying gasp message.

Example

```
Switch(config)# dying-gasp primary snmp-trap secondary syslog
```

3. (Optional) Use the **show dying-gasp {status | packets [snmp-trap | syslog | ethernet-oam]}** command to monitor Dying-Gasp configuration status and packet details, SNMP trap, syslog, and Ethernet OAM messages.

Table 5: Syntax description

Command	Description
dying-gasp	Displays Dying-Gasp information.
status	Displays Dying-Gasp configuration status.
packets	Displays detailed information about the created packets.
snmp-trap	Displays Dying-gasp SNMP trap information.
syslog	Displays Dying-gasp syslog message information.

Command	Description
ethernet-oam	Displays Dying-gasp Ethernet OAM message information. **Note** **ethernet-oam** is not supported on Cisco Catalyst IE9300 Rugged Series Switches .

This is a sample example of the **dying gasp packets** command and its output:

```
Switch# show dying-gasp packets
SNMP Trap packet for server 192.168.0.2, link type IP
interface, via GigabitEthernet1/0/24, local IP address 12.1.1.40
encap type is ARPA, local hardware address 6c03.09e7.23c0
next hop IP address 12.1.1.200, next hop hardware address 6c03.09e7.23c0
Syslog errmsg packet for server 192.168.0.2, link type IP
interface, via GigabitEthernet1/0/24, local IP address 12.1.1.40
encap type is ARPA, local hardware address 6c03.09e7.23c0
next hop IP address 12.1.1.200, next hop hardware address 6c03.09e7.23c0
```

This is a sample output for the **show dying-gasp status** command:

```
Switch# show dying-gasp status
Dying Gasp Configuration
SNMP Trap             Enabled (secondary)
Syslog                Enabled (primary)
```

This is a sample output for the **show dying-gasp packets snmp-trap** command:

```
Switch# show dying-gasp packets snmp-trap
SNMP Trap packet for server 192.168.0.2, link type IP
interface, via GigabitEthernet1/0/24, local IP address 12.1.1.40
encap type is ARPA, local hardware address 6c03.09e7.23c0
next hop IP address 12.1.1.200, next hop hardware address 6c03.09e7.23c0
```

This is a sample output for the **show dying-gasp packets syslog** command:

```
Switch# show dying-gasp packets syslog
Syslog errmsg packet for server 192.168.0.2, link type IP
interface, via GigabitEthernet1/0/24, local IP address 12.1.1.40
encap type is ARPA, local hardware address 6c03.09e7.23c0
next hop IP address 12.1.1.200, next hop hardware address 6c03.09e7.23c0
```

4 Factory Reset

Topics:

- Factory reset procedures
- Restrictions for performing a factory reset
- Secure data wipes
- Configure factory reset
- Feature history for factory reset

Factory reset procedures

Explains how a factory reset erases customer-specific data and restores a device to its original configuration, including details about erased and retained data, and scenarios where a factory reset is used.

A factory reset procedure is a device restoration method that

- erases all customer-specific data stored in a device
- restores the device to its original configuration at the time of shipping, and
- removes configurations, log files, boot variables, core files, and credentials such as FIPS-related keys.

Data erased and retained during factory reset

The factory reset process is used in the following scenarios:

- Return Material Authorization (RMA) for a device: If you have to return a device to Cisco for RMA, remove all the customer-specific data before obtaining an RMA certificate for the device.
- Recovering a compromised device: If the key material or credentials that are stored on a device are compromised, reset the device to the factory configuration, and then reconfigure the device.

During a factory reset, the device reloads and enters ROMMON mode. After the factory reset, the device removes all its environment variables, including the **MAC_ADDRESS** and the **SERIAL_NUMBER** variables, which are required to locate and load the software. Perform a reset in ROMMON mode to automatically set the environment variables. The BAUD rate environment variable returns to its default value after a factory reset. Make sure that the BAUD rate and the console speed are the same at all times. Otherwise, the console becomes unresponsive.

After the system reset in ROMMON mode is complete, add the Cisco IOS image either through an USB or TFTP.

Table 6: Data erased and retained during factory reset

Data erased	Data retained
All Cisco IOS images, including the current boot image	Data from remote field-replaceable units (FRUs)
Crash information and logs	Value of the configuration register.
User data, startup and running configuration, and contents of removable storage devices, such as SERIAL Advanced Technology Attachment (SATA), Solid State Drive (SSD), or USB	–
Credentials such as FIPS-related keys	Credentials such as Secure Unique Device Identifier (SUDI) certificates, and public key infrastructure (PKI) keys.
Onboard Failure Logging (OBFL) logs	Licenses
ROMMON variables added by a user.	–

> **Note**
>
> When you perform a factory reset, none of the memory components on a line card are erased. This means the system software, OBFL, bootloader, and environmental variables on the line card, remain as is.

Restrictions for performing a factory reset

Outlines important restrictions you must consider when performing a factory reset, including limitations on software patch restoration, session behavior, and command support for specific platforms and configurations.

- Software patches, if installed on the device, are not restored after the factory reset process.
- If the **factory-reset** command is issued through a VTY session, the session is not restored after completion of the factory reset process.

Secure data wipes

Explains how secure data wipes remove sensitive information from device storage, ensuring that customer-specific data and operational history are unrecoverable after a factory reset.

A secure data wipe is a device storage sanitization method that

- removes software images, device configuration, software logs, and operational history
- eliminates customer-specific data, including network architecture and design, and
- renders information unrecoverable by following NIST SP 800-88 PURGE guidelines.

Secure data wipe reference information

The **all secure** option in the **factory-reset** command performs data sanitization and securely resets the device. After data sanitization, the device reloads and boots with the software image present in flash.

Secure data wipe feature implements guidelines for media sanitization as described in NIST SP 800-88 Rev. 1. NIST 800-88 is a standard published by the National Institute of Standards and Technology (NIST) that provides guidelines for media sanitization. The PURGE standard within NIST 800-88 specifies methods to render data on storage media unrecoverable using laboratory techniques. When a device is sanitized using NIST 800-88 PURGE method, data cannot be recovered through simple non-invasive data recovery techniques or advanced laboratory techniques.

Configure factory reset

Configure a factory reset to erase all configuration and user data from your Switch, restoring it to its original shipping state. Verify that the Switch reboots and enters ROMmon mode after the reset is complete.

Performing a factory reset erases all configuration and user data from your Switch, restoring it to its original shipping state.

Use this procedure when you need to completely erase all configurations, images, and user data from your switch, such as before decommissioning, repurposing, OR troubleshooting persistent issues.

This process is applicable to standalone, stacked, and StackWise Virtual enabled Switchs.

Before you begin

- Ensure that all the software images, including the current image, configurations, and personal data are backed up before you begin the factory reset process.
- Ensure that there is uninterrupted power supply when the factory reset process is in progress.
- Ensure that In-Service Software Upgrade (ISSU) or In-Service Software Downgrade (ISSD) are not in progress before you begin the factory reset process.

Perform these steps to perform a factory reset:

Procedure

1. Use the **enable** command to enable privileged EXEC mode.

   Example

   ```
   Switch> enable
   ```

   Enter your password, if prompted.

2. Use the **factory-reset** command to perform factory reset to reset the switch to its configuration at the time of its shipping.

   Example

   ```
   Switch# factory-reset all
   ```

   OR

   ```
   Switch# factory-reset switch 1 all config
   ```

   OR

   ```
   Switch# factory-reset all secure
   ```

   The available options are:

   - **all**: Erases all the content from the NVRAM, all the Cisco IOS images, including the current boot image, boot variables, startup and running configuration data, and user data. We recommend that you use this option.
   - **all secure**: Performs data sanitization and securely resets the Switch.

Note

- You can use the **all secure** option only on standalone Switchs.
- This option implements guidelines for media sanitization as described in NIST SP 800-88 Rev. 1.
- The **factory-reset all secure** command initiates data sanitization. The booted image of the Switch is retained.
- When data sanitization is completed, the Switch reloads, and the Switch image is retained in flash if it was booted with an image from the flash.

- **secure 3-pass**: Erases all the content from the Switch with 3-pass overwrite.
  - Pass 1: Overwrites all addressable locations with binary zeroes.
  - Pass 2: Overwrites all addressable locations with binary ones.
  - Pass 3: Overwrites all addressable locations with a random bit pattern.

Note

This option takes approximately thrice the time taken to perform any other option.

- **config**: Resets the startup configurations.
- **boot-vars**: Resets the user-added boot variables.
- **switch** {*switch-number* **all**}:
  - *switch-number*: Specifies the switch number. The range is from 1 to 16.
  - **all**: Selects all the switches in the stack.

Note

The factory reset operation permanently erases all device configurations and security keys (including FIPS keys), restoring the device to its original factory-default state. Reload the switch after the execution of **factory-reset config** command.

After the factory reset process is successfully completed, the switch reboots and enters ROMmon mode.

3. (Optional) Use the **show platform software factory-reset secure log** command to verify the data sanitization report after factory reset.

Example

```
Switch# show platform software factory-reset secure log
Factory reset log:
#CISCO xxxxx DATA SANITIZATION REPORT#
START : 18-09-2022, 06:18:44
END : 18-09-2022, 06:23:36
-MTD-
PNM : nor
NIST : PURGE
-eMMC-
MID : 'Micron'
PNM : 'Q2J55L'
SN : 0x00000001
NIST : PURGE
```

Feature history for factory reset

Identify the Cisco IOS XE release that introduced factory reset support for restoring the switch to its original shipping state and erasing customer-specific data.

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE 26.1.1	Factory reset	Factory reset restores the switch to its original shipping state by erasing customer-specific data, configurations, logs, boot variables, and credentials such as FIPS-related keys.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn.

5 Migration to Meraki Cloud Management Dashboard

Topics:

- Meraki cloud management dashboard migration
- Supported switch models for cloud-managed mode
- How to migrate to a Meraki cloud management dashboard
- How the device behaves during migration

Meraki cloud management dashboard migration

Learn how migration to Meraki cloud-managed mode changes switch management, console access, storage, and configuration.

The Meraki cloud management dashboard is a centralized, web-based platform used to configure and monitor Cisco devices and services. Starting from release 26.2.1, you can migrate the Cisco IE9300 rugged series switch to cloud-managed mode. This migration enables you to manage and control your device through the Meraki cloud-based dashboard.

Key aspects of this migration behaviour and access include:

- After migration, the switch operates entirely under the control of the Meraki cloud management dashboard.
- Console access to the switch is unavailable after migration. To troubleshoot or configure the device, use the Meraki cloud management dashboard.

  If you cannot use the DNS name, you can access the local status page by entering the device's IP address in your browser. This approach helps if you set up the device on a network without Dynamic Host Configuration Protocol (DHCP) or before assigning its final IP address.

  You can use the IP address 1.1.1.100 from the client device to access the local status page (LSP). To access this address, configure a device with these IP settings and browse to the address in a web browser.

  - IP address: 1.1.1.99
  - Subnet mask: 255.255.255.0

- During migration, the flash memory, USB drives, and SD card of the switch are formatted and reconfigured for the Meraki cloud management runtime environment. This process erases all previous configurations and the device must be reconfigured through the Meraki cloud management dashboard.

Supported switch models for cloud-managed mode

Lists the Cisco Industrial Ethernet IE35xx series switch models that support migration to cloud-managed mode.

The following Cisco Industrial Ethernet IE9300 series switch models support management in cloud-managed mode using the Meraki cloud management Dashboard. The minimum Cisco IOS XE software release required for IE9300 platform is specified below.

Platform and release compatibility

The table outlines the minimum Cisco IOS XE software release required for Meraki Dashboard management.

Table 7: Platform and release compatibility

Platform	Minimum Software Release	Dashboard Support Status
IE9300 • IE-9310-26S2C • IE-9320-26S2C • IE-9320-22S2C4X • IE-9320-24P4S • IE-9320-24T4X • IE-9320-24P4X • IE-9320-16P8U4X • IE-9310-16P8S4X	26.2.1	Supported

How to migrate to a Meraki cloud management dashboard

Prepare the switch for Meraki cloud management, validate device compatibility, and register and migrate the device to the dashboard.

Follow these steps to migrate your device to Meraki cloud management dashboard.

Before you begin

- Ensure that you have access to the Meraki cloud management dashboard.
- Ensure the device runs IOS-XE version 26.2.1 or later in install mode.
- Ensure that the uplink interface connecting to the Meraki cloud management dashboard is configured in trunk mode. Also, ensure that the port on the peer device is configured in trunk mode.
- Ensure that the device can resolve DNS names and that it has a routable IP address on the appropriate VLAN via DHCP. For more details, see DHCP IP and DNS configuration on page 66 .
- Ensure that the clock settings (date and time) are configured correctly before onboarding the device. Incorrect settings can cause onboarding or connectivity issues. For more information, see Configure the device clock on page 65.

  Note

  Before initiating the device onboarding process, ensure that the system date and time are configured correctly. Accurate time synchronization is required for certificate validation, secure communication, and event logging. If the system time is not synchronized, authentication errors or onboarding failures may occur.

- Obtain the Cloud ID before adding the device to the network. To register your device and retrieve the Cloud ID, see the Register and migrate to Meraki cloud management dashboard on page 68 section.

Procedure

1. Validate device compatibility. See Validate device compatibility on page 67 for more information.
2. Register and migrate the device. See Register and migrate to Meraki cloud management dashboard on page 68 for more information.

Configure the device clock

Configure the device clock to ensure accurate system time for secure communication and event logging.

Configure the device clock to maintain accurate system time. Accurate system time is required for certificate validation, secure communication, device onboarding, and event logging.

Use Network Time Protocol (NTP) to synchronize the device clock with an external time source. If an NTP server is not available, set the clock manually.

Before you begin

Ensure that the device has network reachability to the NTP server, if you are configuring NTP.

Follow these steps to configure the device clock:

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Switch# configure terminal
   ```

2. Use the **ntp server** *ntp-server-ip-or-hostname* command to configure the NTP server for time synchronization.

   Example

   ```
   Switch# (config)# ntp server 111.0.1.11
   ```

3. Use the **exit** command to return to privileged EXEC mode.

   Example

   ```
   Switch# (config)# exit
   ```

4. (Optional) Use the **clock set** *<hh:mm:ss> <day> <month-name> <year>* command to set the system clock manually when an NTP server is not available. Use this command in privileged EXEC mode.

   Example

   ```
   Switch# clock set 14:30:00 9 July 2026
   ```

5. Use the **show clock** command to verify the system date and time.

Example

```
Switch# show clock
```

DHCP IP and DNS configuration

Perform this procedure to ensure the device can resolve DNS names.

Perform this task to verify that the device can resolve DNS names and that it has a routable IP address on the appropriate VLAN via DHCP. You can also verify that the device routes traffic to external networks through the default gateway. To optimize performance and system reliability, use Cisco-validated SFP modules with Meraki cloud management dashboard hardware. Configure the Cisco Catalyst IE9300 Rugged Series uplink port as a Trunk port to enable VLAN traffic and support network integration.

Before you begin

If your management VLAN exceeds 512, follow the LSP process to change the upstream VLAN to the management VLAN. For more details on LSP, refer Cisco Meraki Local Status Page.

Procedure

1. Use the **ip name-server** *dns-server* command to configure the device to use a DNS name server.

   Example

   ```
   Device(config)# ip name-server 192.0.2.10
   ```

   You can specify up to six DNS servers if needed.
2. Configure a switch virtual interface (SVI) or routed interface with a routable IP address.
   a) Use the **vlan** *vlan-id* command to create the VLAN for cloud connectivity.

      Example

      ```
      Device(config)# vlan 10         ! VLAN used for cloud connectivity
      ```

   b) Use the **exit** command to exit VLAN configuration mode.

      Example

      ```
      Device(config-vlan)# exit
      ```

   c) Use the **interface vlan** *vlan-id* command to enter interface configuration mode for the VLAN.

      Example

      ```
      Device(config)# interface vlan 10
      ```

      Use the same VLAN ID where the DHCP is present.

   d) Use the **ip address dhcp** command to assign an IP address through DHCP.

Example

```
Device(config-if)# ip address dhcp
```

The DHCP provides gateway and DNS information.

e) Use the **no shutdown** command to enable the VLAN interface.

Example

```
Device(config-if)# no shutdown
```

f) Use the **exit** command to exit the interface configuration mode.

Example

```
Device(config-if)# exit
```

g) Use the **ping meraki.com** command to verify that the ping is successful to the Meraki cloud management dashboard.

Example

```
Device# ping meraki.com
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 52.84.205.125, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
Device#
```

Validate device compatibility

Follow this procedure to validate if your device is compatible for migration.

Procedure

Use the **show cloud-mgmt compatibility** command to verify if your device is compatible for migration.

Example

```
Device# show cloud-mgmt  compatibility
=============================================================================================
Cloud-Mgmt Cloud Monitoring: Compatible
=============================================================================================
Cloud-Mgmt Cloud Management:
Compatibility Check Status
---------------------------------------------------------------------------------------------
Boot Mode  INSTALL  - Compatible
---------------------------------------------------------------------------------------------
Switch#  SKU                                  Bootloader Version
Network Modules
---------------------------------------------------------------------------------------------
1        IE-9320-26S2C      - Compatible    26.1.1r         - Compatible
```

```
N/A
------------------------------------------------------------------------------------------
Compatible SKUs: IE-9320-26S2C, IE-9320-22S2C4X, IE-9320-24P4S, IE-9320-24T4X,
 IE-9320-24P4X, IE-9320-16P8U4X
Compatible NMs :

------------------------------------------------------------------------------------------
```

Register and migrate to Meraki cloud management dashboard

Perform this procedure to register your device and migrate to cloud management mode.

Before you begin

Complete these steps before you initiate the migration process. You can also refer How to migrate to a Meraki cloud management dashboard on page 64.

- Back up your device configuration and any relevant local files, including any USB flash drives, and SD cards connected to the device. This is important because all file systems will be formatted once you initiate the migration process.
- Upgrade to Cisco IOS XE 26.2.1 or later in install mode. For details about install mode upgrades, see the Release Notes for Cisco IE9300 Series Switches.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Use the **service cloud-mgmt connect** command to register and initiate migration of your device to cloud-mgmt mode.

   Running this command generates a *cloud-ID*, which is required to claim the switch in the Meraki cloud management dashboard.

   The device contacts resources on the Meraki cloud management dashboard to register. Use the Cloud ID to add each device to the dashboard. After adding the device to a dashboard network with an active uplink and cloud access, it begins converting to Meraki cloud management cloud mode. When this occurs, the device performs a factory reset, which then restricts console access.

   Example

   ```
   Switch(config)# service cloud-mgmt connect
           Starting to connect switch 1
           Successfully connect to Meraki Dashboard
           Switch 1 has been successfully registered
           Received MAC: AABBCCDDEEFF
           Cloud ID: ABCD-ABCD

   Device Registration Status:
   ------------------------------------------------------------------------------------------
   Switch
    Migration
   Num      PID             Serial Number     Cloud ID       Mac Address
      Status       Mode
   ```

```
------------------------------------------------------------------------------------------
1    IE-9320-16P8U4X     ABCD-ABCD         ABCD-ABCD        aaa.bbb.cccc
Registered IE9K-C [Monitoring]

Please note down the Cloud ID as that is required for claiming it in Cloud-Mgmt
 Dashboard
```

Example

```
Switch# show cloud-mgmt
Switch                Serial
Migration
Num    PID            Number           Cloud ID      Mac Address       Status
       Mode
------------------------------------------------------------------------------------------
1   IE-9320-26S2C    FDO2546J16L      ABCD-ABCD      aaa.bbb.cccc
Registered     IE9K-C [Monitoring]
```

3. Use the **exit** command to exit global configuration mode.

Example

```
Device(config)# exit
```

Revert the switch to IOS-XE mode

Contact Meraki cloud management support to migrate your device back to IOS-XE mode.

Consider these key aspects when migrating the device:

- Ensure that the device is online and connected to the Meraki cloud management dashboard before beginning the process.
- The migration process resets the device to its default factory settings and configurations.
- Ensure the migration to IOS-XE mode is complete before deploying the device in production networks.

How the device behaves during migration

Describes the sequence of events that occur on devices during migration to Meraki mode, including storage handling and dashboard integration.

Summary

This section lists the events that occur on the device during migration.

- The device boots up with the default configuration and obtains its operational configuration from the Meraki cloud management dashboard. This migration process usually takes 5 to 10 minutes.
- During the migration process to Meraki cloud management mode, external removable storage such as SD cards and USB drives will be formatted and disabled. These storage devices will not be operational.
- When the migration is complete, you can control your device from the Meraki cloud management dashboard.

Refer to the Configure your Dashboard Network documentation for more information and configuration best practices.

Workflow