



Software-Defined Application Visibility and Control

- [Software-defined application visibility and control, on page 1](#)
- [Enable software-defined application visibility and control on a WLAN \(CLI\), on page 3](#)
- [Configure software-defined application visibility and control global parameters \(CLI\), on page 4](#)
- [List of package files, on page 5](#)
- [Verify SD-AVC controller connection status, on page 5](#)
- [Verify SD-AVC details in AP, on page 5](#)

Software-defined application visibility and control

A software-defined application visibility and control system is a network-level application visibility and control (AVC) controller that

- aggregates application data from multiple devices and sources
- provides composite analytics and telemetry on enterprise network traffic, and
- enables centralized deployment and management of protocol pack updates.

SD-AVC recognizes most enterprise network traffic and provides analytics, visibility, and telemetry into application recognition on the network. SD-AVC profiles all endpoints connected to access nodes, including wireless bridged virtual machines, to perform anomaly detection operations such as Network Address Translation (NAT). SD-AVC alerts you when the same MAC address is used at the same time on different networks.

You can enable the Software-Defined Application Visibility and Control feature on a per-WLAN basis. You can enable or disable Software-Defined Application Visibility and Control functionalities independently.



Note Restart the Capwapd process or reload the AP to resume SD-AVC operation after the SD-AVC process (stilepd) crashes.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 1: Feature history for Software-defined application visibility and control

Feature Name	Release Information	Feature Description
Software-defined application visibility and control (SD-AVC) wireless support with IPv6	Cisco IOS XE 17.18.1	<p>From Cisco IOS XE 17.18.1 onwards, this feature extends the support for adding IPv6 SD-AVC controller or end-point address.</p> <p>These platforms are supported:</p> <ul style="list-style-type: none"> • Cisco Catalyst 9800 controllers: 9800-40, 9800-80, 9800-L, 9800-CL, 9800-SW, CW9800M, CW9800H1, and CW9800H2. • Cisco Catalyst 9300/9400 switches in Fabric mode. • Cisco Wave 2, Wi-Fi 6/6E, and Wi-Fi 7 APs. <p>SD-AVC IPv6 is not supported on Cisco Wireless AireOS Controllers, Cisco Embedded Wireless Controller on Catalyst APs, and Cisco Wave 1 APs.</p>
Software-defined application visibility and control (SD-AVC) wireless support with IPv4	Cisco IOS XE 17.17.1	<p>From Cisco IOS XE 17.17.1 onwards, this feature extends the support for adding the AP and controller payload code for only IPv4 SD-AVC addresses.</p>
Software-defined application visibility and control	Cisco IOS XE 17.7.1	<p>Software-Defined AVC aggregates application data from multiple sources and provides composite application information.</p> <p>These commands are introduced:</p> <ul style="list-style-type: none"> • address and avcsd-service • controller and destination-ports • dscp • segment • source-interface • transport and application-updates • vrf and showsdave ap download status • showsdave status ap

Supported platforms for SD-AVC IPv6

- Cisco Catalyst 9800 controllers—9800-40, 9800-80, 9800-L, 9800-CL, and 9800-SW.

- Cisco Catalyst 9300/9400 switches in Fabric mode.
- Cisco Wave 2 and 11AX APs.

Non supported platforms for SD-AVC IPv6

- Cisco Wireless AireOS Controllers.
- Cisco Embedded Wireless Controller on Catalyst APs.
- Cisco Wave 1 APs

Enable software-defined application visibility and control on a WLAN (CLI)

Allow the system to recognize and manage applications on a WLAN by enabling software-defined application visibility and control using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure WLAN policy profile and enter the wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy test-policy-profile
```

Step 3 Disable central switching and enable local switching.

Example:

```
Device(config-wireless-policy)# no central switching
```

Step 4 Enable application recognition on the wireless policy profile by activating the NBAR2 engine.

Example:

```
Device(config-wireless-policy)# ip nbar protocol-discovery
```

Step 5 Exit wireless policy configuration mode and return to the privileged EXEC mode.

Example:

```
Device(config-wireless-policy)# end
```

Configure software-defined application visibility and control global parameters (CLI)

Enable SD-AVC globally and configure connectivity parameters for SD-AVC controllers using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable SD-AVC and enter the software-definition service configuration mode.

Example:

```
Device(config-sd-service)# avc sd-service
```

Step 3 Configure a segment name identifying a group of devices sharing the same application services.

Example:

```
Device(config-sd-service)# segment AppRecognition
```

Step 4 Enter SD service controller configuration mode to configure connectivity parameters.

Example:

```
Device(config-sd-service)# controller
```

Step 5 Configure controller IP address. Supports only IPv4 address.

Example:

```
Device(config-sd-service-controller)# address 209.165.201.0
```

Step 6 Configure the destination port for communicating with the controller.

Example:

```
Device(config-sd-service-controller)# destination-ports sensor-exporter 21730
```

Step 7 Enable DSCP marking and configure source interface for communicating with the controller.

Example:

```
Device(config-sd-service-controller)# dscp 16
```

```
Device(config-sd-service-controller)# source-interface GigabitEthernet21
```

Step 8 Configure transport protocols for communicating with the controller and associate the VRF with the source interface.

Example:

```
Device(config-sd-service-controller)# transport application-updates https url-prefix cisco
```

```
Device(config-sd-service-controller)# vrf doc-test
```

Step 9 Exit the SD service controller configuration mode and enter the privileged EXEC mode.

Example:

```
Device(config-sd-service-controller)# end
```

List of package files

To view the list of package files list that are sent from the SD-AVC to the controller, use the following command:

```
Device# dir bootflash:sdavc
Directory of bootflash:/sdavc/

251671  -rw-                6728  Jun 26 2025 03:26:54 +00:00  sdavc_config.json
251674  -rw-                 338  Jun 26 2025 03:21:55 +00:00  import_file_meta.json
251670  -rw-                10913  Jun 26 2025 03:20:52 +00:00  PPKD_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack
251669  -rw-                6487  Jun 26 2025 03:20:21 +00:00  pp_update_pp_minor_taxonomy_b72edc9e6ed2e42.json
251672  -rw-                 912  Jun 26 2025 03:20:21 +00:00  pp_update_C9400_HA_a_v2_b30473cb9912.pack
348817  drwx                4096  Jun 26 2025 03:16:27 +00:00  container_application

11250098176 bytes total (3345641472 bytes free)
```

Verify SD-AVC controller connection status

To display the SD-AVC connection status and information summary in the controller, run the **show avc sd-service info summary** command.

```
Device# show avc sd-service info summary
Status: CONNECTED
Device ID: SDA_BOX
Device segment name: C9400_HA
Device address: 192.0.2.1
Device OS version: 17.18.01prd15
Device type: C9407R
Active controller:
  Type      : Primary
  Address   : 192.0.2.1
  Status    : Connected
  Version   : 4.4.0
  Last connection: 06:21:35.000 UTC Thu Jun 19 2025
Active SDAVC import files:
  Protocol pack:          Not loaded
  Secondary protocol pack: PPKD_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack
  Rules pack:            pp_update_C9400_HA_a_v2_b32d0890d8a3.pack
```

Verify SD-AVC details in AP

Verify SD-AVC package files in flash

To verify the protocol files received from the controller, run the **show flash sdavc-pkg** command in the AP.

```

Cisco-AP# show flash sdavc-pkg
Files under /usr/bin/sdavc/
total 36K
drwxr-xr-x   2 Cisco   root           40 Jun 19 06:10 container_application
-rw-r--r--   1 Cisco   root          6487 Jun 19 06:11
pp_update_pp_minor_taxonomy_b72edc9e6ed2e42.json
-rw-r--r--   1 Cisco   root         10922 Jun 19 06:11
PPDK_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack
-rw-r--r--   1 Cisco   root           959 Jun 19 06:11
pp_update_C9400_HA_a_v2_b32d0890d8a3.pack
-rw-r--r--   1 Cisco   root           314 Jun 19 06:11 import_file_meta.json
-rw-r--r--   1 Cisco   root          6728 Jun 19 06:20 sdavc_config.json
-----
Filesystem                Size      Used Available Use% Mounted on
/dev/mmcblk0p32           2.8G      12.7M      2.6G    0% /storage

```

Verify SD-AVC configurations from controller

To verify the SD-AVC configurations sent from the controller, run the **show ap fast-path configuration sd-avc** command in the AP.

```

Cisco-AP# show ap fast-path configuration sd-avc
source IP address       : 0.0.0.0
source port (WLC)      : 9999
destination IPV4 address: 9.2.39.36
destination port (SD-AVC): 50000
gateway MAC             : 00:0C:29:3C:00:8D
ewlc SD-AVC MAC        : 00:00:00:05:00:01
DSCP                    : 0
Fast Path Admin State  : enabled
AP SD-AVC Status       : enabled

```

Verify SD-AVC enablement in AP

To verify the SD-AVC enablement in the AP, run the **show avc status** command.

```

Cisco-AP# show avc status
VAP FNF-STATUS AVC-QOS-STATUS SD AVC-STATUS APM-STATUS
0   Disabled      Disabled      Enabled      Disabled
1   Disabled      Disabled      Enabled      Disabled
2   Disabled      Disabled      Enabled      Disabled
3   Disabled      Disabled      Disabled     Disabled
4   Disabled      Disabled      Disabled     Disabled
5   Disabled      Disabled      Disabled     Disabled
6   Disabled      Disabled      Disabled     Disabled
7   Disabled      Disabled      Disabled     Disabled
8   Disabled      Disabled      Disabled     Disabled
9   Disabled      Disabled      Disabled     Disabled
10  Disabled      Disabled      Disabled     Disabled
11  Disabled      Disabled      Disabled     Disabled
12  Disabled      Disabled      Disabled     Disabled
13  Disabled      Disabled      Disabled     Disabled
14  Disabled      Disabled      Disabled     Disabled
15  Disabled      Disabled      Disabled     Disabled

```