



## Fastlane+

- [Information About Fastlane+, on page 1](#)
- [Configuring an Fastlane+ on a WLAN \(CLI\), on page 1](#)
- [Configuring an Fastlane+ on a WLAN \(GUI\), on page 2](#)
- [Monitoring Fastlane+, on page 2](#)
- [Verifying Fastlane+, on page 3](#)

## Information About Fastlane+

IEEE 802.11ax allows scheduled access-based uplink transmissions by periodically collecting buffer status reports from clients. The Fastlane+ feature improves the effectiveness of estimating the uplink buffer status for clients, thereby enhancing the user experience for latency-sensitive applications. The Fastlane+ feature can be enabled or disabled on a per-WLAN basis. Support for this feature is indicated in the beacons and probe responses transmitted by an AP.



**Note** This feature works only if Protected Management Frame (PMF) is configured as optional or mandatory for a WLAN.

## Configuring an Fastlane+ on a WLAN (CLI)

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b> <b>Example:</b> Device# configure terminal	Enters global configuration mode.
<b>Step 2</b>	<b>wlan profile-name wlan-id SSID_Name</b> <b>Example:</b>	Configures a WLAN and enters WLAN configuration submode.

	Command or Action	Purpose
	Device(config)# wlan wlan-test 3 ssid-test	<b>Note</b> If you have already configured a WLAN, enter the <b>wlan profile-name</b> command.
<b>Step 3</b>	<b>scheduler asr</b>  <b>Example:</b> Device(config-wlan)# scheduler asr	Configures Fastlane+ feature on a WLAN.

## Configuring an Fastlane+ on a WLAN (GUI)

### Procedure

- 
- Step 1** Choose **Configuration > Tags & Profiles > WLANs**.
  - Step 2** Select a WLAN.
  - Step 3** Click **Advanced** tab.
  - Step 4** Check the **Advanced Scheduling Requests Handling** check box to enable the feature on a per-WLAN basis.
  - Step 5** Click **Update & Apply to Device**.
- 

## Monitoring Fastlane+

### Procedure

- 
- Step 1** Choose **Monitoring > Wireless > Clients**.
  - Step 2** Click a client name from the client list.  
The **Client** window with multiple tabs is activated.
  - Step 3** Click **General** tab.
  - Step 4** Click **Client Statistics** tab.  
The most recent uplink latency statistics received from the client is displayed in the **Uplink Latency Distribution** section.
  - Step 5** Click **Client Properties** tab.  
The Fastlane+ feature-related client capabilities information is displayed at the bottom of the window.
-

## Verifying Fastlane+

The following example shows how to verify whether Fastlane+ is enabled or disabled for a WLAN:

```
Device# show wlan 2 | include ASR

Advanced Scheduling Requests Handling : Enabled
```

The following example shows how to verify Fastlane+ capability information and the most recent client uplink latency statistics:

```
Device# show wireless client mac-address f45c.89b0.xxxx detail
.
.
.
Regular ASR support: : ENABLED
Non-default Fastlane Profile: : Active
Range Voice Video Background Best-Effort
-----
[0-20ms] 400 300 200 100
[20-40ms] 401 301 201 101
[40-100ms] 402 302 202 102
[>100ms] 403 303 203 103
```

The following example shows how to verify Fastlane+ statistics along with Fastlane+ capability and uplink latency statistics for all the Fastlane+ clients on a WLAN.



**Note** `show interfaces dot11radio asr-info all` is an AP command, and does not work on the controller.

```
Device# show interfaces Dot11Radio 1 asr-info all

[*10/12/2020 18:45:21.0149]
[*10/12/2020 18:45:21.0150] Client-MAC:[26:52:CF:C8:D0:1C] AID:[3] ASR-Capability:[0x1]
[*10/12/2020 18:45:21.0150] BE- LAT[0-20]:[267] LAT[20-40]:[57] LAT[40-100]:[32]
LAT[>100]:[26]
[*10/12/2020 18:45:21.0150] BK- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0] LAT[>100]:[0]
[*10/12/2020 18:45:21.0150] VI- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0] LAT[>100]:[0]
[*10/12/2020 18:45:21.0150] VO- LAT[0-20]:[2222] LAT[20-40]:[409] LAT[40-100]:[224]
LAT[>100]:[163]
[*10/12/2020 18:45:21.0150]
[*10/12/2020 18:45:21.0206] HTT_PEER_DETAILS_TLV:
[*10/12/2020 18:45:21.0206] peer_type = 0
[*10/12/2020 18:45:21.0206] sw_peer_id = 98
[*10/12/2020 18:45:21.0206] vdev_id = 25
[*10/12/2020 18:45:21.0206] pdev_id = 0
[*10/12/2020 18:45:21.0206] ast_idx = 1187
[*10/12/2020 18:45:21.0206] mac_addr = 26:52:cf:c8:d0:1c
[*10/12/2020 18:45:21.0206] peer_flags = 0x200006f9
[*10/12/2020 18:45:21.0206] qpeer_flags = 0x8
[*10/12/2020 18:45:21.0206]
[*10/12/2020 18:45:21.0206] HTT_STATS_PEER_ASR_STATS_TLV
[*10/12/2020 18:45:21.0206] asr_bmap: 0x8
[*10/12/2020 18:45:21.0206] asr_muedca_update_cnt: 1
[*10/12/2020 18:45:21.0206] asr_muedca_reset_cnt: 1
[*10/12/2020 18:45:21.0206] asr_ul_mu_bsr_trigger: 2376
[*10/12/2020 18:45:21.0206] asr_min_trig_intv- BE:0 BK:0 VI:0 VO:19
[*10/12/2020 18:45:21.0206] asr_max_trig_intv- BE:0 BK:0 VI:0 VO:20
[*10/12/2020 18:45:21.0207] asr_min_alloc_rate- BE:0 BK:0 VI:0 VO:12
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_cnt- BE:0 BK:0 VI:0 VO:2149
```

```

[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_bytes- BE:0          BK:0 VI:0 VO:757546
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_cnt- BE:0          BK:0 VI:0 VO:5002
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_bytes- BE:0          BK:0 VI:0 VO:2400960
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_cnt- BE:0          BK:0 VI:0 VO:2134
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_bytes- BE:0          BK:0 VI:0 VO:736578
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_padding_bytes- BE:0          BK:0 VI:0 VO:2953488

```

The following examples show how to verify scheduling statistics along with capability and uplink latency statistics for a given client on a WLAN:



**Note** The `show interfaces dot11radio asr-info` is an AP command and it will not work on the controller.

```

Device# show interfaces Dot11Radio 1 asr-info 26:XX:CF:XX:D0:XX

[*10/12/2020 18:45:21.0149]
[*10/12/2020 18:45:21.0150] Client-MAC:[26:52:CF:C8:D0:1C] AID:[3] ASR-Capability:[0x1]
[*10/12/2020 18:45:21.0150] BE- LAT[0-20]:[267] LAT[20-40]:[57] LAT[40-100]:[32]
LAT[>100]:[26]
[*10/12/2020 18:45:21.0150] BK- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0] LAT[>100]:[0]
[*10/12/2020 18:45:21.0150] VI- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0] LAT[>100]:[0]
[*10/12/2020 18:45:21.0150] VO- LAT[0-20]:[2222] LAT[20-40]:[409] LAT[40-100]:[224]
LAT[>100]:[163]
[*10/12/2020 18:45:21.0150]
[*10/12/2020 18:45:21.0206] HTT_PEER_DETAILS_TLV:
[*10/12/2020 18:45:21.0206] peer_type = 0
[*10/12/2020 18:45:21.0206] sw_peer_id = 98
[*10/12/2020 18:45:21.0206] vdev_id = 25
[*10/12/2020 18:45:21.0206] pdev_id = 0
[*10/12/2020 18:45:21.0206] ast_idx = 1187
[*10/12/2020 18:45:21.0206] mac_addr = 26:xx:cf:xx:d0:xx
[*10/12/2020 18:45:21.0206] peer_flags = 0x200006f9
[*10/12/2020 18:45:21.0206] qpeer_flags = 0x8
[*10/12/2020 18:45:21.0206]
[*10/12/2020 18:45:21.0206] HTT_STATS_PEER_ASR_STATS_TLV
[*10/12/2020 18:45:21.0206] asr_bmap: 0x8
[*10/12/2020 18:45:21.0206] asr_muedca_update_cnt: 1
[*10/12/2020 18:45:21.0206] asr_muedca_reset_cnt: 1
[*10/12/2020 18:45:21.0206] asr_ul_mu_bsr_trigger: 2376
[*10/12/2020 18:45:21.0206] asr_min_trig_intv- BE:0          BK:0 VI:0 VO:19
[*10/12/2020 18:45:21.0206] asr_max_trig_intv- BE:0          BK:0 VI:0 VO:20
[*10/12/2020 18:45:21.0207] asr_min_alloc_rate- BE:0          BK:0 VI:0 VO:12
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_cnt- BE:0          BK:0 VI:0 VO:2149
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_bytes- BE:0          BK:0 VI:0 VO:757546
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_cnt- BE:0          BK:0 VI:0 VO:5002
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_bytes- BE:0          BK:0 VI:0 VO:2400960
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_cnt- BE:0          BK:0 VI:0 VO:2134
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_bytes- BE:0          BK:0 VI:0 VO:736578
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_padding_bytes- BE:0          BK:0 VI:0 VO:2953488

```