Aerohive Networks Inc.

Background scan test case

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 1.0 | 2009/06/11 | LiangfuZhang | Initial version |
| 1.1 | 2009/10/10 | Liangfu | Add one case and detail test step to test case |
| 1.2 | 2010/07/26 | Tiezhu Zhu | Add cases for HT40/ Europe mode and ACSP mode. |
| 1.3 | 2011/03/19 | Tiezhu Zhu | Add test case for backhaul/dual mode and add test case for DFS channel |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

We do background scan for radio in access mode periodically. Non-voice packets are buffered in QoS module when background scan is on going. Following is the way to do background scan:

1. Don’t scan all the channels in one shot. Just scan one channel at a time and go back to working channel immediately.
2. When a background scan is kicked, start with the first channel.
3. Every second check if we can scan. If can scan, scan a channel for a short time (~50ms) and come back to working channel.
4. Next second scan the next channel if allowed.
5. Keep doing it until reaching the last channel.

Following are prerequisites to do a background scan:

1. Whether any VOIP call is on going, only do scan if no VOIP call is going on.
2. Get the overall traffic load of the radio, only do scan if load is below a certain threshold.

We allow user to do following configuration for background scan:

1. Enable/disable background scan
2. Interval between each full scan
3. Whether allow background scan when there’s traffic with voice priority.

40M-above mode? Europe country code channel? 3-channel model?

Active scan and passive scan at the same time?

Do not do Bgscan when power save mode

# Test point or strategy

## Configure managerment

### CLI check: [no] radio profile <name> scan access

**pass-yypan**

### CLI check: [no] radio profile <name> scan access interval xx

**PASS**

**pass-yypan**

### CLI check: [no] radio profile <name> scan access client xx

**PASS**

**pass-yypan**

### CLI check: [no] radio profile <name> scan access voice

## check Bgscan function

### Check bgscan default configuration

**pass-yypan**

### Check probe request packets on all channels with Ominipeek

### Check if box send out CTS packets in current channel when box do bgscan.

**pass-yypan**

PASS

### Check the debug information

**pass-yypan**

### Check the scan interval

**pass-yypan**

### When there is voip call, skip the scan

### When disable voip, shouldn’t skip scan

### Check if box do scan when there is client.

PASS Default scan.

### When overall load exceed the limit, skip the scan

Add one upload test case

**pass-yypan**

### When Client upload data to AP, check if AP do BG scan after overall load exceed the limit.

(50% load threshold for Cario 2011-03-19)

### Check if AP can find all neighbor with 7 boxes in screen room

When neighbor AP is under backhaul mode, BG scan can not find it.

PASS

### When neighbor AP is under backhaul mode, check if BG scan will find it.

PASS

**pass-yypan**

### Check if AP can find all neighbors in office

### When radio mode is backhaul/dual mode, box also can do BGScan. (2011-03-19 Tiezhu)

Why not do bgscan? Any side effect?

Pass (2011-03-19 Tiezhu)

**pass-yypan**

### Check if BG scan work normally when ACSP channel select conflict with BG scan

### Check if BG scan work normally when ACSP power change conflict with BG scan

### Under Europe mode, check if BG scan all channels

PASS

**pass-yypan**

### Under HT40-Above/FCC mode, check if BG scan will scan all channels

PASS

**pass-yypan**

### Under HT40-Below/FCC mode, check if BG scan will scan all channels

**pass-yypan**

### Under HT40-Above/Europe mode, check if BG scan will scan all channels

Pass

**pass-yypan**

### Under HT-40-Below/Europe mode, Check if BG scan will scan all channels

**pass-yypan**

### Under 3 channels mode/FCC, check if BG scan will scan all channels.

**pass-yypan**

### Under 4 channels mode/FCC, check if BG scan will scan all channels.

**pass-yypan**

### 

2.2.24 Under mesh state, check if AP will do BGScan

**pass-yypan**

**do BGScan**

2.2.25 Under mesh state and mp also have mmp connected, check if mp will do BGscan? If disable client, check if mp will do Bgscan

2.2.26 Under HT40 above/below mode, check if BGscan mark ACSP table list with HT20/40-above/HT40-below mode

2.2.27 When BGscan enable, check if IDP scan work normally?

2.2.28 Under DFS channel, check if BGscan will do passive scan in DFS channel?

Fail

2.2.29 Under DFS channel, check if ACSP table update after BGScan.

2.2.30 During BGscan, check if VAP state variable

2.2.31 Duration: Enable BGScan and run traffic for 24 hours, check client connected issue. (Bgscan interval 1 minutes)

2.2.32 When client in power save state, check if BGscan work?

2.2.33 BGscan probe request times and probe response.

Test next time. It is hard to send probe response in first test.

2.2.34： BGscan is base on radio, check BGscan work if some VAP shutdown.

Ssid+dual

## Performance

### Set scan interval to 1 minute, do performance test.

## Stability

### Set scan interval to minimum (1 minute), check client stability

## HM case

# Topology

AP under test + many acsp neighbors

# Hardware and software needed

Screen room

Ominipeek and wireless adapter

802.11abgn dula

# TestCase

## CLI check: [no] radio profile <name> scan access

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | bgscan \_Cli\_1 | | |
| Priority | Accept | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Check CLI: [no] radio profile <name> scan access | | |
| Pre-condition |  | | |
| Test procedure | 1.config cli:radio profile <name> scan access  2.config cli:no radio profile <name> scan access | | |
| Expect result | 1.bgscan is enabled by default,so if cli:radio profile <name> scan access can be configured,and no error message,that’s ok!  2.after config cli: no radio profile <name> scan access,it can be found by show running config. | | |

## CLI check: [no] radio profile <name> scan access interval xx

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | bgscan \_Cli\_2 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Check CLI: [no] radio profile <name> scan access interval xx | | |
| Pre-condition |  | | |
| Test procedure | 1.config cli: radio profile <name> scan access interval xx,xx Range:  1-1440,Default: 10 minutes.  2.config cli:no radio profile <name> scan access interval xx. | | |
| Expect result | 1.cli can be found by show running config.  2.after config this cli, cli:radio profile <name> scan access interval xx will not found by show running config. | | |

## CLI check: [no] radio profile <name> scan access client

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | bgscan \_Cli\_3 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Check CLI: [no] radio profile <name> scan access client | | |
| Pre-condition |  | | |
| Test procedure | 1.config cli: radio profile <name> scan access client  2.config cli: no radio profile <name> scan access client | | |
| Expect result | 1.this cli is configured by default,so if it can be configured and no error message,that’s ok!  2.after configured this cli,it can be found by show running config. | | |

## CLI check: [no] radio profile <name> scan access voice

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | bgscan \_Cli\_4 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Check CLI: [no] radio profile <name> scan access voice | | |
| Pre-condition |  | | |
| Test procedure | 1.config cli: radio profile <name> scan access voice  2.config cli: no radio profile <name> scan access voice | | |
| Expect result | 1.cli can be found by show running config.  2.after config cli: no radio profile <name> scan access voice,cli:radio profile <name> scan access voice will not be found by show running config. | | |

## check default bgscan configuration

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_1 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check default bgscan configuration | | |
| Pre-condition |  | | |
| Test procedure | 1.reset configure and reboot box  2.check wifi0/wifi1 bgscan status by show int wifi0/wifi1  3.create new radio profile  4.check bgscan status by show radio profile <name> | | |
| Expect result | 2.bgcan is enabled,scan interval is 10 minutes.  4. bgcan is enabled,scan interval is 10 minutes. | | |

## check probe request packets on all channels with ominipeek

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | One AP | | |
| Description | check probe request packets on all channels with ominipeek | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan  2.capture packets in channel 1  3.change channel to 2 and capture packets with ominipeek  4.repeat step 4 until all channel is tested | | |
| Expect result |  | | |

## check if box send out CTS packets in current channel when do bgscan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_3 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check if box send out CTS packets in current channel when do bgscan | | |
| Pre-condition | creat a radio profile ng, phymode 11ng,bind to wifi0.  Creat a ssid,bind to wifi0. | | |
| Test procedure | 1.enable bgscan on wifi0.  2.set bgscan interval 1 minute.  3.open \_kdebug wifi-driver wifi0.1 scan  4.show logging buffered | include self | | |
| Expect result | 4. box send out CTS packets in current channel when do bgscan | | |

## check the debug information

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_4 | | |
| Priority | Middle | Automation Flag | No |
| Topology to use | One AP | | |
| Description | check the debug information | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  3.the debug information:  a.passive scan or active scan  b.dwell time | | |
| Expect result |  | | |

## check scan interval

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_5 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check scan interval | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to certain value  3.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  4.check if the scan interval is correct. | | |
| Expect result | Box should do bgscan according to the interval. | | |

## when there is voip call, skip the scan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_6 | | |
| Priority | High | Automation Flag | No |
| Topology to use | One AP | | |
| Description | Check when there is voip call, skip the scan | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.Excute: radio profile <name> scan access voice  3.set scan interval to certain value  4.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  5.generate voip traffic, check if box still do bgscan.  qos classifier-profile aa interface/ssid  ssid test\_cwj\_11 qos-classifier aa  qos classifier-map ssid test\_cwj\_11 6 | | |
| Expect result | when there is voip call, skip the scan. | | |

## when disable voip, box shouldn’t skip scan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_7 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | when disable voip, box shouldn’t skip scan | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.Excute: no radio profile <name> scan access voice  3.set scan interval to certain value  4.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  5.generate voip traffic, check if box skip scan | | |
| Expect result | when disable voip, box shouldn’t skip scan | | |

## check if box do bgscan when there is client

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_8 | | |
| Priority | High | Automation Flag | No |
| Topology to use | One AP | | |
| Description | check if box do bgscan when there is client | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.Excute: radio profile <name> scan access client  3.set scan interval to certain value  4.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  5.connect laptop to wifi0, check if box skip scan  6.excute: no radio profile <name> scan access client  7.connect laptop to wifi0, check if box do bgscan | | |
| Expect result | box will do bgscan when config “radio profile <name> scan access client”  box will not do bgscan when config “radio profile <name> scan access client” | | |

## check if box do bgscan when interface load exceed certain limit

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_9 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check if box do bgscan when interface load exceed certain limit | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to certain value  3.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  4.run downlink performance test, check if box skip scan | | |
| Expect result | box will not do bgscan when interface load exceed certain limit | | |

## When Client upload data to AP, check if AP do BG scan after overall load exceed the limit.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_10 | | |
| Priority | High | Automation Flag | No |
| Topology to use | One AP | | |
| Description | When Client upload data to AP, check if AP do BG scan after overall load exceed the limit. | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to certain value  3.open bgscan debug:  \_kdebug wifi-driver wifi0.1 scan  Debug console  4.run uplink performance test, check if box skip scan | | |
| Expect result |  | | |

## check if box can find all neighbors in screen room

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_11 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check if box can find all neighbors in screen room | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.put 6 bx in screen room (box under access mode)  3.set scan interval to 1 minute  4.set wifi0/wifi1 power and channel to fixed value  5.reboot box  6.check if box can find all neighbors in screen room | | |
| Expect result | 1.box should find all neighbors. | | |

## When neighbor AP is under backhaul mode in screen room, check if BG scan can find it

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_12 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | When neighbor AP is under backhaul mode in screen room, check if BG scan can find it | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.put 2 bx in screen room (box under backhaul mode)  3.set scan interval to 1 minute  4.set wifi0/wifi1 power and channel to fixed value  5.reboot box  6.check if box can find all neighbors in screen room | | |
| Expect result | BG scan can find backhaul neighbor. | | |

## check if box can find all neighbors in office

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_13 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check if box can find all neighbors in office | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.put 6 bx in screen room  3.set scan interval to 1 minute  4.set wifi0/wifi1 power and channel to fixed value  5.reboot box  6.check if box can find all neighbors in office | | |
| Expect result | box can find all neighbors in office | | |

## check box bgscan when radio mode is backhaul

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_14 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | check box bgscan when radio mode is backhaul | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi1  2.set wifi1 to backhaul mode  3.open debug:  \_kdebug wifi-driver wifi1 scan  Debug console  4.check if box scan channel. | | |
| Expect result | Box will do bgscan and all wifi1 channel will be scaned. | | |

## set scan interval to 1 minute, check client stability

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_15 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | set scan interval to 1 minute, check client stability | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to 1 nimute  3.connect laptop to AP and generate ping traffic  4.check if the connection is stable. | | |
| Expect result |  | | |

## Check if BG scan work normally when ACSP channel select conflict with BG scan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_16 | | |
| Priority | High | Automation Flag | No |
| Topology to use | One AP | | |
| Description | Check if BG scan work normally when ACSP channel select conflict with BG scan | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to 1 nimute  3.Open debug mode: \_kdebug wifi-driver wifi0.1 scan” “debug console”  4.Set static channel for AP wifi0: in wifi0 radio channel 6  5.Set static channel to auto mode while do BG scan. | | |
| Expect result | BG scan and ACSP scan can make effect one by one | | |
| Test result |  | | |

## Under Europe/826 mode, check if BG scan all channels

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_17 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under Europe/826 mode, check if BG scan all channels | | |
| Pre-condition | set country mode to Europe mode 826,and reboot AP. | | |
| Test procedure | 1.enable bgscan on wifi0.  2.set scan interval to 1 minute.  3.Open debug mode: \_kdebug wifi-driver wifi0.1 scan” “debug console”  4.Check if BG scan all wifi0 channels  5.enable bgscan on wifi1.  6.open dfs channel.  7.set scan interval to 1 minute  8.Open debug mode: \_kdebug wifi-driver wifi1.1 scan” “debug console”  9.Check if BG scan all wifi1 channels. | | |
| Expect result | 4.BG scan all wifi0 channel.  9.Check if BG scan all wifi1 channels,include dfs channel. | | |
| Test result |  | | |

## Check if BG scan work normally when ACSP power select conflict with BGscan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_18 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Check if BG scan work normally when ACSP power select conflict with BG scan | | |
| Pre-condition |  | | |
| Test procedure | 1.enable bgscan on wifi0  2.set scan interval to 1 nimute  3.Open debug mode: \_kdebug wifi-driver wifi0.1 scan” “debug console”  4.Set static power for AP wifi0: in wifi0 radio power 5  5.Set static power to auto mode while do BG scan. | | |
| Expect result | BG scan and ACSP power can make effect one by one | | |

## Under HT40-Above/FCC mode, check if BG scan will scan all channels

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_19 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under HT40-Above/FCC mode, check if BG scan will scan all channels | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code fcc  2.Set wifi0 channel-width 40-Above  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel.  6.set wifi1 channel-width 40-above  7.enable bgscan on wifi1.  8.set wifi1 bgscan interval to 1 minute.  9.check if bgscan will scan all wifi1 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel.  9.bgscan will scan all wifi1 channel. | | |

## Under HT40-Below/FCC mode, check if BG scan will scan all channels

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_20 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under HT40-Below/FCC mode, check if BG scan will scan all channels | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code fcc  2.Set wifi0 channel-width 40-below  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel.  6.set wifi1 channel-width 40-below  7.enable bgscan on wifi1.  8.set wifi1 bgscan interval to 1 minute.  9.check if bgscan will scan all wifi1 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel.  9.bgscan will scan all wifi1 channel. | | |

## Under HT40-Above/Europe mode, check if BG scan will scan all channels

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_21 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under HT40-Above/Europe mode, check if BG scan will scan all channels | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code world/826  2.Set wifi0 channel-width 40-Above  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel.  6.set wifi1 channel-width 40-above  7.enable bgscan on wifi1.  8.set wifi1 bgscan interval to 1 minute.  9.check if bgscan will scan all wifi1 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel.  9.bgscan will scan all wifi1 channel. | | |

## Under HT-40-Below/Europe mode, Check if BG scan will scan all channels

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_22 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under HT-40-Below/Europe mode, Check if BG scan will scan all channels | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code world/826  2.Set wifi0 channel-width 40-below  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel.  6.set wifi1 channel-width 40-below  7.enable bgscan on wifi1.  8.set wifi1 bgscan interval to 1 minute.  9.check if bgscan will scan all wifi1 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel.  9.bgscan will scan all wifi1 channel. | | |

## Under 3 channels mode/FCC, check if BG scan will scan all channels.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_23 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under 3 channels mode/FCC, check if BG scan will scan all channels. | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code fcc.  2.Set wifi0 channel-mode 3-channels.  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel. | | |

## Under 4 channels mode/FCC, check if BG scan will scan all channels.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_24 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | One AP | | |
| Description | Under 4 channels mode/FCC, check if BG scan will scan all channels. | | |
| Pre-condition |  | | |
| Test procedure | 1.Set country code fcc.  2.Set wifi0 channel-mode 4-channels.  3.enable bgscan on wifi0.  4.set wifi0 bgscan interval to 1 minute.  5.check if bgscan will scan all wifi0 channel. | | |
| Expect result | 5.bgscan will scan all wifi0 channel. | | |

## Under mesh state, check if AP will do BGScan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_25 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | -----------------------------  | |  AP1 AP2 | | |
| Description | Under mesh state, check if AP will do BGScan. | | |
| Pre-condition |  | | |
| Test procedure | 1.AP1 wifi1 and AP2 wifi1 backhual mode.  2.shutdown AP2 eth0.  3.after AP2 mesh with AP1,enable bgscan on AP2 wifi1.  4.set bgscan interval to 1 minute.open \_kdebug wifi-driver w1.1 scan  5.check if AP2 will do bgscan. | | |
| Expect result | AP2 will do bgscan. | | |

## Under mesh state and mp also have mmp connected, check if mp will do BGscan? If disable client, check if mp will do Bgscan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_26 | | |
| Priority | High | Automation Flag | No |
| Topology to use | --------------------------------------------------  | | |  AP1 AP2 AP3 | | |
| Description | Under mesh state and mp also have mmp connected, check if mp will do BGscan? If disable client, check if mp will do Bgscan. | | |
| Pre-condition |  | | |
| Test procedure | 1.AP1 wifi1 and AP2 wifi1 AP3 wifi1 backhual mode.  2.shutdown AP2 eth0.AP3 eth0.  3.after AP2 mesh with AP1,AP2 mesh with AP1,enable bgscan on AP2 wifi1.  4.set bgscan interval to 1 minute.open \_kdebug wifi-driver w1.1 scan  5.check if AP2 will do bgscan. | | |
| Expect result | AP2 will do bgscan. | | |

## Under HT40 above/below mode, check if BGscan mark ACSP table list with HT20/40-above/HT40-below mode.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_27 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | ----------------------------------------------  | | |  AP1 AP2 AP3 | | |
| Description | Under HT40 above/below mode, check if BGscan mark ACSP table list with HT20/40-above/HT40-below mode. | | |
| Pre-condition |  | | |
| Test procedure | 1.AP1 wifi0 channel-width 40-above,wifi1 channel-width 40-below.AP2 wifi0 channel-width 20  2.enable bgscan on AP3 wifi0 and wifi1.  3.set AP3 bgscan interval 1 minute.  4.show acsp \_nbr on AP3, check if BGscan mark ACSP table list with HT20/40-above/HT40-below mode. | | |
| Expect result | 4. BGscan mark ACSP table list with HT20/40-above/HT40-below mode. | | |

## When BGscan enable, check if IDP scan work normally.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_28 | | |
| Priority | High | Automation Flag | No |
| Topology to use |  | | |
| Description | When BGscan enable, check if IDP scan work normally. | | |
| Pre-condition |  | | |
| Test procedure |  | | |
| Expect result |  | | |

## Under DFS channel, check if BGscan will do passive scan in DFS channel.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_29 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use |  | | |
| Description | Under DFS channel, check if BGscan will do passive scan in DFS channel. | | |
| Pre-condition |  | | |
| Test procedure | 1.AP wifi1 select a dfs channel.  2.enable bgscan on wifi1,set interval 1 minute.  3.open \_kdebug wifi-driver wifi1.1 scan  4.check if BGscan will do passive scan in DFS channel. | | |
| Expect result | 4. BGscan will do passive scan in DFS channel. | | |

## Under DFS channel, check if ACSP table update after BGscan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_30 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | -----------------------------------  | |  AP1 AP2 | | |
| Description | Under DFS channel, check if ACSP table update after BGScan. | | |
| Pre-condition |  | | |
| Test procedure | 1.two ap in screen room.  2.enable ap2 wifi1 bgscan.set interval to 1 minute.  3.change ap1 wifi1 dual mode to access.  4.show acsp \_nbr on ap2 check if ap2 update ap1 wifi1 mode. | | |
| Expect result | Ap2 will update ap1 wifi1 mode. | | |

## During BGscan, check if VAP state variable

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_31 | | |
| Priority | High | Automation Flag | No |
| Topology to use | -----------------------------------  | |  AP1 AP2 | | |
| Description | During BGscan, check if VAP state variable | | |
| Pre-condition |  | | |
| Test procedure | 1.two ap in screen room.  2.enable ap2 wifi1 bgscan.set interval 1 minute.  3.shutdown ap1 wifi1,and no shutdown.ap1 wifi1 will do acsp.  4.show acsp \_nbr on ap2,ap1’s state will transfer from init->listen->run. | | |
| Expect result | 4.ap1 wifi1 state variable. | | |

## Enable BGScan and run traffic for 24 hours, check client connected issue. (Bgscan interval 1 minutes)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_32 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Client1------AP~~~~client2 | | |
| Description | Enable BGScan and run traffic for 24 hours, check client connected issue. (Bgscan interval 1 minutes) | | |
| Pre-condition |  | | |
| Test procedure | 1.client1 connect to eth0.  2.client2 connect AP wifi1 by ssid.  3.ap enable bgscan on wifi1.  4.two client run traffic for 24 hours.  5.check client connect issue. | | |
| Expect result | 5.client2 always connect to ap | | |

## When client in power save state, check if BGscan work

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_33 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use | AP~~~~~~client | | |
| Description | When client in power save state, check if BGscan work | | |
| Pre-condition |  | | |
| Test procedure | 1.enable power save.  “radio profile <string> scan access client power-save”  2.enable wifi1 bgscan.set interval 1 minute.  3.check if bgscan work. | | |
| Expect result | 3.allow scanning to occur when connected clients are in a power save state | | |

## BGscan probe request times and probe response.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_34 | | |
| Priority | High | Automation Flag | No |
| Topology to use |  | | |
| Description | BGscan probe request times and probe response. | | |
| Pre-condition |  | | |
| Test procedure | 1.enable wifi1 bgscan,set interval 1 minute.  2.capture packets,check probe request. | | |
| Expect result |  | | |

## BGscan is base on radio, check BGscan work if some VAP shutdown.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Bgscan\_Function\_35 | | |
| Priority | High | Automation Flag | Yes |
| Topology to use |  | | |
| Description | BGscan is base on radio, check BGscan work if some VAP shutdown. | | |
| Pre-condition |  | | |
| Test procedure | 1.wifi0 access mode,bind 3 ssid.  2.wifi1 dual mode.  3.enable bgscan on wifi0,set interval 1 minute.  4.enable bgscan on wifi1,set interval 1 minute.  5.shutdown 2 ssid on wifi0,check if wifi0 do bgscan.  6.shutdown wifi1.1,check if wifi1 do bgscan. | | |
| Expect result | 5.wifi0 do bgscan.  6.wifi1 do bgscan. | | |

**Dual band bgscan test case:**

## check default radio profile radio\_ng0 bgscan state.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_1 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check default radio profile radio\_ng0 bgscan state. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) show radio profile radio\_ng0. | | |
| Expect result | BGSCAN allow=enabled; BGSCAN during voice=disabled;  BGSCAN interval=10 minutes; BGSCAN CTS-to-Self=enabled;  BGSCAN with client=enabled; BGSCAN with PS client=disabled; | | |
| Test result |  | | |
| Comment |  | | |

## creat a new 11ng and 11na radio profile,check the default bgscan state.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_2 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | creat a new 11ng and 11na radio profile,check the default bgscan state. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) show radio profile ng. | | |
| Expect result | BGSCAN allow=enabled; BGSCAN during voice=disabled;  BGSCAN interval=10 minutes; BGSCAN CTS-to-Self=enabled;  BGSCAN with client=enabled; BGSCAN with PS client=disabled; | | |
| Test result |  | | |
| Comment |  | | |

## check if bgscan can be disabled and enabled.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_3 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check if bgscan can be disabled and enabled. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) disable bgscan in radio profile ng.  4) show radio profile ng.result 1.  5) enable bgscan in radio profile ng.  6) show radio profile ng.result 2. | | |
| Expect result | Result 1: BGSCAN allow=disabled  Result 2: BGSCAN allow=enabled | | |
| Test result |  | | |
| Comment |  | | |

## check if bgscan interval can be modified.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_4 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check if bgscan interval can be modified. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) modify bgscan interval to 1 minute.  4) show radio profile ng.result 1.  5) revocation interval in radio profile ng.  6) show radio profile ng.result 2. | | |
| Expect result | Result 1: BGSCAN interval=1 minute  Result 2: BGSCAN interval=10 minutes | | |
| Test result |  | | |
| Comment |  | | |

## Check CLI: [no] radio profile <name> scan access client.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_5 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | Check CLI: [no] radio profile <name> scan access client. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) config cli: radio profile ng scan access client.  4) show radio profile ng.result 1.  5) config cli: no radio profile ng scan access client.  6) show radio profile ng.result 2. | | |
| Expect result | Result 1: BGSCAN with client=enabled  Result 2: BGSCAN with client=disabled | | |
| Test result |  | | |
| Comment |  | | |

## Check CLI: [no] radio profile <name> scan access voice

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_6 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | Check CLI: [no] radio profile <name> scan access voice. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) config cli: radio profile ng scan access voice.  4) show radio profile ng.result 1.  5) config cli: no radio profile ng scan access voice.  6) show radio profile ng.result 2. | | |
| Expect result | Result 1: BGSCAN with client=enabled  Result 2: BGSCAN with client=disabled | | |
| Test result |  | | |
| Comment |  | | |

## Check CLI: [no] radio profile <name> scan access client power-save

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_cli\_7 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | Check CLI: [no] radio profile <name> scan access client power-save. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a new radio profile ng,phymode 11ng.  3) config cli: radio profile ng scan access client power-save.  4) show radio profile ng.result 1.  5) config cli: no radio profile ng scan access client power-save.  6) show radio profile ng.result 2. | | |
| Expect result | Result 1: BGSCAN with PS client=enabled;  Result 2: BGSCAN with PS client=disabled; | | |
| Test result |  | | |
| Comment |  | | |

## check bgscan interval

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_1 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check bgscan interval | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a ssid,bind to wifi0.  3) config cli:  \_kdebug wifi-driver w0.1 scan  debug console  4) wait 25minutes,show logging buffered | include self  5) check the scan interval between first scan and second scan.result 1.  6) creat a new radio profile ng,phymode 11ng,bind to wifi0.  7) set bgscan interval I minute.  8) wait 4 minutes, show logging buffered | include self  9) check the scan interval between first scan and second scan.result 2. | | |
| Expect result | Result 1: 10 minutes  Result 2: 1 minute. | | |
| Test result |  | | |
| Comment |  | | |

## check if box do bgscan after bgscan disabled.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_2 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check if box do bgscan after bgscan disabled. | | |
| Pre-condition |  | | |
| Test procedure | 1) reset box.  2) creat a radio profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) config cli:  \_kdebug wifi-driver w0.1 scan  debug console  4) set bgscan interval 1 minute.  5) disable bgscan.  6) wait 2 minutes, show logging buffered | include self.  7) check if box do bgscan.result 1. | | |
| Expect result | Result 1: box will not do bgscan. | | |
| Test result |  | | |
| Comment |  | | |

## check bgscan channel list and if it is passive scan in dfs channel.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_3 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check bgscan channel list and if it is passive scan in dfs channel | | |
| Pre-condition |  | | |
| Test procedure | 1) set region fcc.and reboot box.  2) creat a new profile ng,phymode 11ng,bind to wifi0.  3) creat a ssid,bind to wifi0.  4) set bgscan interval 1 minute.  5) wait 2 minutes,show logging buffered | include self.  6) check bgscan channel list.result 1.  7) show logging buffered | include self.  8) check if passive in dfs channel.result 2.  9) set region world,country-code 826.save config,reboot box.  10) wait 2 minutes,show logging buffered | include self.  11) check bgscan list.result 3.  12) show logging buffered | include self.  13) check if passive in dfs channel.result 4. | | |
| Expect result | Result 1:36~64,100~116,132~140,149~165.  Result 2:passive scan in dfs channel.  Result 3:36~64,100~140.  Result 4:passive scan in dfs channel. | | |
| Test result |  | | |
| Comment | Fcc:  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh lo bu | in passive  1970-01-01 00:05:33 debug kernel: 33.532 | scanner\_state\_foreign\_channel\_entry: entering channel=140 passive=1 probe\_delay=0  1970-01-01 00:05:33 debug kernel: 33.302 | scanner\_state\_foreign\_channel\_entry: entering channel=136 passive=1 probe\_delay=0  1970-01-01 00:05:33 debug kernel: 33.072 | scanner\_state\_foreign\_channel\_entry: entering channel=132 passive=1 probe\_delay=0  1970-01-01 00:05:32 debug kernel: 32.842 | scanner\_state\_foreign\_channel\_entry: entering channel=116 passive=1 probe\_delay=0  1970-01-01 00:05:32 debug kernel: 32.612 | scanner\_state\_foreign\_channel\_entry: entering channel=112 passive=1 probe\_delay=0  1970-01-01 00:05:32 debug kernel: 32.382 | scanner\_state\_foreign\_channel\_entry: entering channel=108 passive=1 probe\_delay=0  1970-01-01 00:05:32 debug kernel: 32.152 | scanner\_state\_foreign\_channel\_entry: entering channel=104 passive=1 probe\_delay=0  1970-01-01 00:05:31 debug kernel: 31.922 | scanner\_state\_foreign\_channel\_entry: entering channel=100 passive=1 probe\_delay=0  1970-01-01 00:05:31 debug kernel: 31.692 | scanner\_state\_foreign\_channel\_entry: entering channel=165 passive=0 probe\_delay=0  1970-01-01 00:05:31 debug kernel: 31.462 | scanner\_state\_foreign\_channel\_entry: entering channel=161 passive=0 probe\_delay=0  1970-01-01 00:05:31 debug kernel: 31.232 | scanner\_state\_foreign\_channel\_entry: entering channel=157 passive=0 probe\_delay=0  1970-01-01 00:05:31 debug kernel: 31.002 | scanner\_state\_foreign\_channel\_entry: entering channel=153 passive=0 probe\_delay=0  1970-01-01 00:05:30 debug kernel: 30.772 | scanner\_state\_foreign\_channel\_entry: entering channel=149 passive=0 probe\_delay=0  1970-01-01 00:05:30 debug kernel: 30.544 | scanner\_state\_foreign\_channel\_entry: entering channel=10 passive=0 probe\_delay=0  1970-01-01 00:05:30 debug kernel: 30.316 | scanner\_state\_foreign\_channel\_entry: entering channel=9 passive=0 probe\_delay=0  1970-01-01 00:05:30 debug kernel: 30.088 | scanner\_state\_foreign\_channel\_entry: entering channel=8 passive=0 probe\_delay=0  1970-01-01 00:05:29 debug kernel: 29.860 | scanner\_state\_foreign\_channel\_entry: entering channel=5 passive=0 probe\_delay=0  1970-01-01 00:05:29 debug kernel: 29.632 | scanner\_state\_foreign\_channel\_entry: entering channel=4 passive=0 probe\_delay=0  1970-01-01 00:05:29 debug kernel: 29.404 | scanner\_state\_foreign\_channel\_entry: entering channel=3 passive=0 probe\_delay=0  1970-01-01 00:05:29 debug kernel: 29.176 | scanner\_state\_foreign\_channel\_entry: entering channel=2 passive=0 probe\_delay=0  1970-01-01 00:05:28 debug kernel: 28.742 | scanner\_state\_foreign\_channel\_entry: entering channel=48 passive=0 probe\_delay=0  1970-01-01 00:05:28 debug kernel: 28.512 | scanner\_state\_foreign\_channel\_entry: entering channel=44 passive=0 probe\_delay=0  1970-01-01 00:05:28 debug kernel: 28.282 | scanner\_state\_foreign\_channel\_entry: entering channel=40 passive=0 probe\_delay=0  1970-01-01 00:05:28 debug kernel: 28.052 | scanner\_state\_foreign\_channel\_entry: entering channel=36 passive=0 probe\_delay=0  1970-01-01 00:05:27 debug kernel: 27.822 | scanner\_state\_foreign\_channel\_entry: entering channel=64 passive=1 probe\_delay=0  1970-01-01 00:05:27 debug kernel: 27.592 | scanner\_state\_foreign\_channel\_entry: entering channel=60 passive=1 probe\_delay=0  1970-01-01 00:05:27 debug kernel: 27.362 | scanner\_state\_foreign\_channel\_entry: entering channel=56 passive=1 probe\_delay=0  1970-01-01 00:05:27 debug kernel: 27.132 | scanner\_state\_foreign\_channel\_entry: entering channel=52 passive=1 probe\_delay=0  1970-01-01 00:05:26 debug kernel: 26.904 | scanner\_state\_foreign\_channel\_entry: entering channel=7 passive=0 probe\_delay=0  1970-01-01 00:05:26 debug kernel: 26.677 | scanner\_state\_foreign\_channel\_entry: entering channel=11 passive=0 probe\_delay=0  1970-01-01 00:05:26 debug kernel: 26.449 | scanner\_state\_foreign\_channel\_entry: entering channel=6 passive=0 probe\_delay=0  1970-01-01 00:05:26 debug kernel: 26.221 | scanner\_state\_foreign\_channel\_entry: entering channel=1 passive=0 probe\_delay=0  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#  World 826:  AH-8d0f00#sh lo bu | in passive  1970-01-01 00:04:24 debug kernel: 4294931.703 | scanner\_state\_foreign\_channel\_entry: entering channel=140 passive=1 probe\_delay=0  1970-01-01 00:04:24 debug kernel: 4294931.473 | scanner\_state\_foreign\_channel\_entry: entering channel=136 passive=1 probe\_delay=0  1970-01-01 00:04:23 debug kernel: 4294931.243 | scanner\_state\_foreign\_channel\_entry: entering channel=132 passive=1 probe\_delay=0  1970-01-01 00:04:23 debug kernel: 4294931.013 | scanner\_state\_foreign\_channel\_entry: entering channel=128 passive=1 probe\_delay=0  1970-01-01 00:04:23 debug kernel: 4294930.783 | scanner\_state\_foreign\_channel\_entry: entering channel=124 passive=1 probe\_delay=0  1970-01-01 00:04:23 debug kernel: 4294930.553 | scanner\_state\_foreign\_channel\_entry: entering channel=120 passive=1 probe\_delay=0  1970-01-01 00:04:23 debug kernel: 4294930.323 | scanner\_state\_foreign\_channel\_entry: entering channel=116 passive=1 probe\_delay=0  1970-01-01 00:04:22 debug kernel: 4294930.093 | scanner\_state\_foreign\_channel\_entry: entering channel=112 passive=1 probe\_delay=0  1970-01-01 00:04:22 debug kernel: 4294929.861 | scanner\_state\_foreign\_channel\_entry: entering channel=108 passive=1 probe\_delay=0  1970-01-01 00:04:22 debug kernel: 4294929.631 | scanner\_state\_foreign\_channel\_entry: entering channel=104 passive=1 probe\_delay=0  1970-01-01 00:04:22 debug kernel: 4294929.397 | scanner\_state\_foreign\_channel\_entry: entering channel=100 passive=1 probe\_delay=0  1970-01-01 00:04:21 debug kernel: 4294929.169 | scanner\_state\_foreign\_channel\_entry: entering channel=12 passive=0 probe\_delay=0  1970-01-01 00:04:21 debug kernel: 4294928.941 | scanner\_state\_foreign\_channel\_entry: entering channel=10 passive=0 probe\_delay=0  1970-01-01 00:04:21 debug kernel: 4294928.713 | scanner\_state\_foreign\_channel\_entry: entering channel=9 passive=0 probe\_delay=0  1970-01-01 00:04:21 debug kernel: 4294928.485 | scanner\_state\_foreign\_channel\_entry: entering channel=8 passive=0 probe\_delay=0  1970-01-01 00:04:20 debug kernel: 4294928.257 | scanner\_state\_foreign\_channel\_entry: entering channel=5 passive=0 probe\_delay=0  1970-01-01 00:04:20 debug kernel: 4294928.029 | scanner\_state\_foreign\_channel\_entry: entering channel=4 passive=0 probe\_delay=0  1970-01-01 00:04:20 debug kernel: 4294927.801 | scanner\_state\_foreign\_channel\_entry: entering channel=3 passive=0 probe\_delay=0  1970-01-01 00:04:20 debug kernel: 4294927.573 | scanner\_state\_foreign\_channel\_entry: entering channel=2 passive=0 probe\_delay=0  1970-01-01 00:04:20 debug kernel: 4294927.337 | scanner\_state\_foreign\_channel\_entry: entering channel=48 passive=1 probe\_delay=0  1970-01-01 00:04:19 debug kernel: 4294927.105 | scanner\_state\_foreign\_channel\_entry: entering channel=44 passive=1 probe\_delay=0  1970-01-01 00:04:19 debug kernel: 4294926.866 | scanner\_state\_foreign\_channel\_entry: entering channel=40 passive=1 probe\_delay=0  1970-01-01 00:04:19 debug kernel: 4294926.633 | scanner\_state\_foreign\_channel\_entry: entering channel=36 passive=1 probe\_delay=0  1970-01-01 00:04:19 debug kernel: 4294926.403 | scanner\_state\_foreign\_channel\_entry: entering channel=64 passive=1 probe\_delay=0  1970-01-01 00:04:18 debug kernel: 4294926.173 | scanner\_state\_foreign\_channel\_entry: entering channel=60 passive=1 probe\_delay=0  1970-01-01 00:04:18 debug kernel: 4294925.943 | scanner\_state\_foreign\_channel\_entry: entering channel=56 passive=1 probe\_delay=0  1970-01-01 00:04:18 debug kernel: 4294925.709 | scanner\_state\_foreign\_channel\_entry: entering channel=52 passive=1 probe\_delay=0  1970-01-01 00:04:18 debug kernel: 4294925.481 | scanner\_state\_foreign\_channel\_entry: entering channel=13 passive=0 probe\_delay=0  1970-01-01 00:04:17 debug kernel: 4294925.253 | scanner\_state\_foreign\_channel\_entry: entering channel=7 passive=0 probe\_delay=0  1970-01-01 00:04:17 debug kernel: 4294925.025 | scanner\_state\_foreign\_channel\_entry: entering channel=11 passive=0 probe\_delay=0  1970-01-01 00:04:17 debug kernel: 4294924.793 | scanner\_state\_foreign\_channel\_entry: entering channel=6 passive=0 probe\_delay=0  1970-01-01 00:04:17 debug kernel: 4294924.566 | scanner\_state\_foreign\_channel\_entry: entering channel=1 passive=0 probe\_delay=0 | | |

## check if ap can find 11ng+access mode 20M/40M neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_4 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap can find 11ng+access mode 20M/40M neighbor. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap 11ng+access+20M,wait 5 minute.  2) show acsp neighbor on box.result 1.  3) neighbor ap 11ng+access+40M, wait 5 minute.  4) show acsp neighbor on box.result 2. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list.  Result 2:neighbor vap In acsp neighbor list. | | |
| Test result |  | | |
| Comment | 1) 11ng+access+20M  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:8990 Access 1 1 -39 yes 6 0 0 40+  0019:7745:9710 Access 1 5 -19 yes 5 0 0 20  0019:7745:4cd0 Access cathy 1 -30 yes 7 0 0 20  0019:7745:4ce0 Backhaul hive0 48 -36 yes 0 0 0 20  0019:7745:9720 Backhaul hive0 36 -29 yes 0 0 0 20  2) 11ng+access+40-above  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:8990 Access 1 1 -39 yes 6 0 0 40+  0019:7745:9710 Access 1 5 -19 yes 5 0 0 20  0019:7745:4cd0 Access cathy 1 -30 yes 7 0 0 20  0019:7745:4ce0 Backhaul hive0 48 -36 yes 0 0 0 20  0019:7745:9720 Backhaul hive0 36 -29 yes 0 0 0 20  3) 11ng+access+40-below  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:8990 Access 1 11 -39 yes 2 0 0 40-  0019:7745:9710 Access 1 5 -19 yes 7 0 0 20  0019:7745:4cd0 Access cathy 1 -31 yes 7 0 0 20  0019:7745:4ce0 Backhaul hive0 48 -36 yes 0 0 0 20  0019:7745:9720 Backhaul hive0 36 -29 yes 0 0 0 20 | | |

## check if ap can find 11ng+backhual mode 20M/40M neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_5 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap can find 11ng+backhaul mode 20M/40M neighbor. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap 11ng+backhaul+20M, wait 5 minute.  2) show acsp neighbor on box.result 1.  3) neighbor ap 11ng+backhaul+40M, wait 5 minute.  4) show acsp neighbor on box.result 2. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list.  Result 2:neighbor vap In acsp neighbor list. | | |
| Test result |  | | |
| Comment | 1) 11ng+backhaul+20M  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:8990 Backhaul hive0 6 -41 yes 0 0 0 20  0019:7745:9710 Access 1 5 -19 yes 9 0 0 20  0019:7745:4cd0 Access cathy 1 -31 yes 6 0 0 20  0019:7745:4ce0 Backhaul hive0 48 -36 yes 0 0 0 20  0019:7745:9720 Backhaul hive0 36 -29 yes 0 0 0 20  2) 11ng+backhaul+40-above  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:8990 Backhaul hive0 6 -35 yes 4 0 0 40+  3) 11ng+backhaul+40-below  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:4cd0 Access cathy 1 -34 yes 8 0 3 20  0019:7745:9720 Backhaul hive0 36 -20 yes 2 0 0 20  0019:7745:4ce0 Backhaul hive0 153 -39 yes 1 0 0 20  0019:7703:8990 Backhaul hive0 11 -28 yes 2 0 0 40- | | |

## check if ap can find 11na+access mode 20M/40M neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_6 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap can find 11na+access mode 20M/40M neighbor. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap 11na+access+20M, wait 5 minute.  2) show acsp neighbor on box.result 1.  3) neighbor ap 11na+access+40M, wait 5 minute.  4) show acsp neighbor on box.result 2. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list.  Result 2:neighbor vap In acsp neighbor list. | | |
| Test result |  | | |
| Comment | 1) 11na+access+20M  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:4cd0 Access cathy 1 -36 yes 10 0 0 20  0019:7745:9720 Access 1 40 -20 yes 1 0 0 20  0019:7703:8990 Backhaul hive0 6 -27 yes 3 0 0 40+  2) 11na+access+40-above  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:89a0 Access 1 36 -45 yes 2 0 0 40+  0019:7745:4cd0 Access cathy 1 -32 yes 2 0 3 20  0019:7745:9720 Access 1 40 -21 yes 1 0 0 20    3) 11na+access+40-below  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:4cd0 Access cathy 1 -35 yes 6 0 3 20  0019:7745:9720 Access 1 40 -19 yes 0 0 0 20  0019:7703:89a0 Access 1 48 -45 yes 0 0 0 40-  0019:7745:4ce0 Backhaul hive0 153 -37 yes 0 0 0 20 | | |

## check if ap can find 11na+backhaul mode 20M/40M neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_7 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap can find 11na+backhaul mode 20M/40M neighbor. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap 11na+backhaul+20M, wait 5 minute.  2) show acsp neighbor on box.result 1.  3) neighbor ap 11na+backhaul+40M, wait 5 minute.  4) show acsp neighbor on box.result 2. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list.  Result 2:neighbor vap In acsp neighbor list. | | |
| Test result |  | | |
| Comment | 1) 11na+backhaul+20M  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:89a0 Backhaul hive0 36 -51 yes 0 0 0 20  0019:7745:4cd0 Access cathy 1 -32 yes 4 0 3 20  0019:7745:9720 Access 1 40 -22 yes 1 0 0 20  0019:7703:8990 Backhaul hive0 6 -30 yes 3 0 0 40+  2) 11na+backhaul+40-above  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:89a0 Backhaul hive0 36 -50 yes 1 0 0 40+  0019:7745:4cd0 Access cathy 1 -32 yes 5 0 2 20  0019:7745:9720 Access 1 40 -22 yes 1 0 0 20  0019:7703:8990 Backhaul hive0 6 -30 yes 3 0 0 40+  3) 11na+backhaul+40-below  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7703:89a0 Backhaul hive0 48 -50 yes 0 0 0 40-  0019:7745:4cd0 Access cathy 1 -33 yes 5 0 2 20  0019:7745:9720 Access 1 40 -22 yes 0 0 0 20  0019:7703:8990 Backhaul hive0 6 -30 yes 3 0 0 40+  0019:7745:4ce0 Backhaul hive0 153 -38 yes 0 0 0 20 | | |

## check if ap update neighbor state after neighbor ap change mode.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_8 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap update neighbor state after neighbor ap change mode. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap access 20M->access 40M.wait 5 minutes.  2) show acsp neighbor on box.result 1.  3) neighbor ap access 20M->backhual 20M.wait 5 minutes.  4) show acsp neighbor on box.result 1.  5) neighbor ap access 20M->backhual 40M.wait 5 minutes.  6) show acsp neighbor on box.result 1.  7) neighbor ap backhaul 20M->access 20M.wait 5 minutes.  8) show acsp neighbor on box.result 1.  9) neighbor ap backhaul 20M->access 40M.wait 5 minutes.  10) show acsp neighbor on box.result 1.  11) neighbor ap backhaul 20M->backhual 40M.wait 5 minutes.  12) show acsp neighbor on box.result 1. | | |
| Expect result | Result 1:neighbor vap mode update in acsp neighbor list. | | |
| Test result |  | | |
| Comment |  | | |

## Box radio 40-above mode,check if it can find 40-below mode neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_9 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | Box radio 40-above mode,check if it can find 40-below mode neighbor. | | |
| Pre-condition | put box in screen room. | | |
| Test procedure | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute.  4) set channel-width 40-above.  5) neighbor box wifi0 and wifi1 channel 40-below.  6) wait 5 minutes,show acsp neighbor,check if box can find neighbor vap.result 1.  7) Creat a new profile na,phymode 11na,bind to wifi0.  8) set channel-width 40-above.  9) set bgscan interval 1 minute.  10) neighbor box wifi0 and wifi1 channel 40-below.  11) wait 5 minutes,show acsp neighbor,check if box can find neighbor vap.result 1. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list. | | |
| Test result |  | | |
| Comment | 1) radio 11na mode,channel 40-above  AH-8d0f00#  AH-8d0f00#sh int  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:778d:0f00 - U - 1 - hive0 -  Eth0 0019:778d:0f00 wan U - - - - -  Eth1 0019:778d:0f02 access D - - - hive0 -  Eth2 0019:778d:0f03 access D - - - hive0 -  Eth3 0019:778d:0f04 access D - - - hive0 -  Eth4 0019:778d:0f05 access D - - - hive0 -  Wifi0 0019:778d:0f10 access U 36 - na - -  Wifi0.1 0019:778d:0f10 access U 36 - na hive0 1  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Disable(User disable) 36 Disable(User disable) 14  AH-8d0f00#  AH-8d0f00#sh ra pr na  AC=access category; be=best-effort; bg=background; vi=video; vo=voice;  AIFS=Arbitration Inter-Frame space; Txoplimit=transmission opportunity limit;  WMM=Wi-Fi Multimedia; ACK=acknowledgment; min=minimum; max=maximum;  BGSCAN=background scan; PS=Power save; LB=load balance;  A-MPDU=Aggregate MAC protocol data unit; DFS=Dynamc Frequency Selection;  Radio profile=na;  Phy-mode=11na; Short preamble mode=enabled;  DFS=disabled; A-MPDU=enabled; Short guard interval=disabled;  Channel width=40MHz; Deny client=none;  Transmit chain=static 3; Receive chain=static 3;  Transmit rate=auto; Beacon period=100; Max clients number=100;  AC=be; WMM min CW=4; max CW=6; AIFS=3; txoplimit=0; No ACK mode=disabled;  AC=bg; WMM min CW=4; max CW=10; AIFS=7; txoplimit=0; No ACK mode=disabled;  AC=vi; WMM min CW=3; max CW=4; AIFS=1; txoplimit=3008; No ACK mode=disabled;  AC=vo; WMM min CW=2; max CW=3; AIFS=1; txoplimit=1504; No ACK mode=disabled;  AP channel switch=once only;  Backhaul channel switch=once only;  Interference-Switch:  Interference threshold=25%  CRC error threshold=25%  Max ACSP Tx Power=20dBm;  BGSCAN allow=enabled; BGSCAN during voice=disabled;  BGSCAN interval=1 minutes; BGSCAN CTS-to-Self=enabled;  BGSCAN with client=enabled; BGSCAN with PS client=disabled;  Backhaul failover=enable; Trigger time=2;  Inteference map=disable; Short term interval=5 minutes; Channel utilization threshold=20%; CRC error threshold=20%;  High density=Disable  Safety net=Enable; Band steering=Disable; Load balance=Disable;  Weak SNR threshold=15 dBm; safety net timeout=60 seconds;  LB hold time=60 seconds; LB station airtime limit=4%;  LB interference utilization=40%;LB crc error limit=30%;  LB Neighbor Load Info Query Interval=60s;  Detect Bssid Spoofing=Disabled;  Tx beamforming=Disabled;  Bind interfaces=Wifi0  AH-8d0f00#  AH-8d0f00#sh int w0 ch  Channel 36 : 5180 Mhz 11na 20 40U Channel 40 : 5200 Mhz 11na 20 40L  Channel 44 : 5220 Mhz 11na 20 40U Channel 48 : 5240 Mhz 11na 20 40L  Channel 149 : 5745 Mhz 11na 20 40U Channel 153 : 5765 Mhz 11na 20 40L  Channel 157 : 5785 Mhz 11na 20 40U Channel 161 : 5805 Mhz 11na 20 40L  Channel 165 : 5825 Mhz 11na 20  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:4cd0 Access cathy 1 -34 yes 6 0 2 20  0024:d7f4:3114 Access cathy 1 -27 no -- -- -- --  0019:7745:9720 Access 1 40 -22 yes 0 0 0 20  0019:7745:4ce0 Backhaul hive0 153 -41 yes 0 0 0 20  0019:7703:8990 Backhaul hive0 11 -32 yes 2 0 0 40-  0019:7703:89a0 Backhaul hive0 48 -42 yes 0 0 0 40- | | |

## Check if box can find all platform neighbor.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_10 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | Check if box can find all platform neighbor. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) put ap110/120/170/320/330/BR100 in screen room.  2) creat a vap on this ap.  3) wait 5 minutes,show acsp neighbor,check if all vap in neighbor list and mode correct. | | |
| Expect result | Result 1:neighbor vap In acsp neighbor list.mode correct. | | |
| Test result |  | | |
| Comment |  | | |

## Check if box can find different vendor neighbor

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_11 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | Check if box can find different vendor neighbor. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) put box outdoor.  2) wait 10 minutes,show acsp neighbor,check box list different vendor neighbor. | | |
| Expect result |  | | |
| Test result |  | | |
| Comment | 1)  AH-8d0f00#sh acsp nei | ex 0019  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  08ea:4421:8c51 Access PPSK-Register 1 -42 yes 56 0 0 20  08ea:4422:d1e2 Access driver-1 149 -56 yes 19 0 1 20  008c:1022:1120 Access 157 -73 no -- -- -- --  000b:6b7d:10f0 Access 161 -63 no -- -- -- --  00-8C-10 (hex) Black Box Corp.  008C10 (base 16) Black Box Corp.  1000 Park Drive  Lawrence PA 15055  UNITED STATES  00-0B-6B (hex) Wistron Neweb Corp.  000B6B (base 16) Wistron Neweb Corp.  No. 10-1, Li-Hsin Road I, Science-based  Hsinchu 300  TAIWAN, REPUBLIC OF CHINA | | |

## check if ap update neighbor state while neighbor ap do acsp.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_12 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if ap update neighbor state while neighbor ap do acsp. | | |
| Pre-condition | put box in screen room.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) neighbor ap wifi0 shutdown/no shutdown.  2) show acsp neighbor,check if box update neighbor vap state.result 1.  3) set neighbor wifi0 static channel. wait 5 minutes.  4) show acsp neighbor,check if box update neighbor vap state.result 2. | | |
| Expect result | Result 1:neighbor vap state update in acsp neighbor list.init->listen->run.  Result 2:neighbor vap state should update to static. | | |
| Test result |  | | |
| Comment | 1) static:  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:1 A 00010080 STATIC 51 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 44 73:1 A 00010100 RUN 49 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:1 MA 00040100 STATIC 49 20 0 0 hive0 cefca154 0  2) init->scan->listen->run  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:12 A 00010080 STATIC 21 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 74:34 A 00010100 INIT 0 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 139 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:2 MA 00010100 RUN 376 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 65:1 MA 00040080 STATIC 381 20 0 0 hive0 cefca154 0  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:12 A 00010080 STATIC 26 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 74:80 A 00010100 SCAN 0 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 143 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:2 MA 00010100 RUN 381 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 65:1 MA 00040080 STATIC 385 20 0 0 hive0 cefca154 0  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:12 A 00010080 STATIC 34 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 74:100 A 00010100 SCAN 6 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 152 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:2 MA 00010100 RUN 390 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 65:1 MA 00040080 STATIC 394 20 0 0 hive0 cefca154 0  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:13 A 00010080 STATIC 35 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 74:100 A 00010100 SCAN 67 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 213 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:3 MA 00010100 RUN 31 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 67:2 MA 00040080 STATIC 35 20 0 0 hive0 cefca154 0  AH-8d0f00#  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:13 A 00010080 STATIC 37 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 74:111 A 00010100 LISTEN 0 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 215 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:3 MA 00010100 RUN 33 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 67:2 MA 00040080 STATIC 37 20 0 0 hive0 cefca154 0  AH-8d0f00#  AH-8d0f00#sh acsp \_nbr  wifi0(14) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7745:4cd0 0019:7745:4cd0 0019:7745:4cc0 10.155.35.10 1 64:14 A 00010080 STATIC 4 20 0 0 cathy 0  0019:7745:9720 0019:7745:9720 0019:7745:9700 172.16.91.3 36 73:376 A 00010100 RUN 0 20 0 0 1 0  0019:7703:89a0 0019:7703:89a0 0019:7703:8980 172.16.91.11 48 56:2 MA 00040100 STATIC 242 20 0 0 hive0 cefca154 0  0019:7745:4ce0 0019:7745:4ce0 0019:7745:4cc0 10.155.35.10 153 56:3 MA 00010100 RUN 59 20 0 0 hive0 cefca154 0  0019:7703:8990 0019:7703:8990 0019:7703:8980 172.16.91.11 11 67:2 MA 00040080 STATIC 64 20 0 0 hive0 cefca154 0 | | |

## generate voip traffic,check if box do bgscan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_13 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | generate voip traffic,check if box do bgscan | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) generate voip traffic. | | |
| Expect result | Result 1:box will not do bgscan when there is voice traffic. | | |
| Test result |  | | |
| Comment |  | | |

## config cli:radio profile xxx scan access voice,generate voip traffic,check if box do bgscan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_14 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | config cli:radio profile xxx scan access voice,generate voip traffic,check if box do bgscan. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute.  4) radio profile ng scan access voice. | | |
| Test procedure | 1) generate voip traffic. | | |
| Expect result | Result 1:box will do bgscan when there is voice traffic after config this cli. | | |
| Test result |  | | |
| Comment |  | | |

## check if box do bgscan when there is a client.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_15 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if box do bgscan when there is a client. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) connect a client to ap.  2) wait 5 minute,show logging buffered | include self,check box do bgscan | | |
| Expect result | Result 1:box will do bgscan when there is client. | | |
| Test result |  | | |
| Comment |  | | |

## config cli:no radio profile xxx scan access client.check if box do bgscan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_16 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | config cli:no radio profile xxx scan access client.check if box do bgscan. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute.  4) config cli:no radio profile ng scan access client. | | |
| Test procedure | 1) connect a client to ap.  2) wait 5 minute,show logging buffered | include self,check box do bgscan. result 1. | | |
| Expect result | Result 1:box will not do bgscan when there is client after config this cli. | | |
| Test result |  | | |
| Comment |  | | |

## config cli:radio profile xxx scan access client.check if box do bgscan when client in a power save state.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_17 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | config cli:radio profile xxx scan access client.check if box do bgscan when client in a power save state. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute.  4) config cli:radio profile ng scan access client. | | |
| Test procedure | 1) connect a client to ap.make client go to power-save state.  2) wait 5 minute,show logging buffered | include self,check box do bgscan. result 1. | | |
| Expect result | Result 1:box will not do bgscan when there client in power-save state. | | |
| Test result |  | | |
| Comment |  | | |

## config cli:radio profile xxx scan access client power-save.check if box do bgscan when client in a power save state.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_18 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | config cli:radio profile xxx scan access client power-save.check if box do bgscan when client in a power save state. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute.  4) config cli:radio profile ng scan access client power-save. | | |
| Test procedure | 1) connect a client to ap.make client go to power-save state.  2) wait 5 minute,show logging buffered | include self,check box do bgscan. result 1. | | |
| Expect result | Result 1:box will do bgscan when there client in power-save state after config this cli. | | |
| Test result |  | | |
| Comment |  | | |

## check if box do bgscan when radio load exceed limit.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_19 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check if box do bgscan when radio load exceed limit. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat a ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) config cli: \_test interface wifi0 cont-tx 1000000.  2) config cli:  \_debug dcd chnl  Debug console  2) wait 5 minute,show logging buffered | include self,check box do bgscan. result 1. | | |
| Expect result | Result 1:box will skip bgscan when radio load exceed limit. | | |
| Test result |  | | |
| Comment |  | | |

## check if ap do bgscan when all vaps down and some vaps down.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_20 | | |
| Priority | Middle | Automation Flag | Yes |
| Topology to use |  | | |
| Description | check if ap do bgscan when all vaps down and some vaps down. | | |
| Pre-condition | 1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) creat 6 ssid,bind to wifi0.  3) set bgscan interval 1 minute. | | |
| Test procedure | 1) showdown all ssids.  2) config cli:  \_kdebug wifi-driver w0.1 scan  \_kdebug wifi-driver w0.2 scan  \_kdebug wifi-driver w0.3 scan  \_kdebug wifi-driver w0.4 scan  \_kdebug wifi-driver w0.5 scan  \_kdebug wifi-driver w0.6 scan  Debug console  2) wait 5 minute,show logging buffered | include self,check box do bgscan. result 1.  3) no shutdown 1 ssid.  4) wait 5 minute,show logging buffered | include self,check box do bgscan. result 2. | | |
| Expect result | Result 1:box will not do bgscan since all vaps down.  Result 2:box will do bgscan since 1 vap up. | | |
| Test result |  | | |
| Comment |  | | |

## Enable BGScan and run traffic for 24 hours, check client connected issue.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_21 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | Enable BGScan and run traffic for 24 hours, check client connected issue. | | |
| Pre-condition |  | | |
| Test procedure | 1) use default radio profile,creat a ssid bind to wifi0.  2) connect a client to ap.  3) ap as iperf server,client as iperf client.run traffic for 24hours between client and ap.  4) show station,check A-Time.result 1  5) creat a new radio profile,bgscan interval 1 minute,bind to wifi0.  6) connect a client to ap.  7) ap as iperf server,client as iperf client.run traffic for 24hours between client and ap.  8) show station,check A-Time.result 1. | | |
| Expect result | Result 1: A-Time last 24 hours. | | |
| Test result |  | | |
| Comment |  | | |

## dual band bgscan should not work on other platform.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_22 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | dual band bgscan should not work on other platform.(Ap120/170/320/330/BR100) | | |
| Pre-condition |  | | |
| Test procedure | Repeat this step on Ap120/170/320/330/BR100.  1) creat a radio profile ng,phymode 11ng,bgscan interval 1 minute,bind on wifi0.  2) config cli:\_kdebug wifi-driver w0.1 scan  Debug console  3) wait 2 minutes,show logging buffered | include self,check scan channel list.result 1.  Repeat this step on Ap120/170/320/330  4) creat a radio profile ng,phymode 11na,bgscan interval 1 minute,bind on wifi1.  5) config cli:\_kdebug wifi-driver w1.1 scan  Debug console  6) wait 2 minutes,show logging buffered | include self,check scan channel list.result 2. | | |
| Expect result | Result 1:bgscan should only scan wifi0 channel.  Result 2:bgscan should only scan wifi1 channel. | | |
| Test result |  | | |
| Comment |  | | |

## Check dual/backhaul mode dual band scan.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_23 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use |  | | |
| Description | check dual/backhaul mode dual band scan. | | |
| Pre-condition | 1) region fcc,and reboot.  1) Creat a new profile ng,phymode 11ng,bind to wifi0.  2) set bgscan interval 1 minute. | | |
| Test procedure | 1) wifi0 backhual mode.  2) wait 2 minutes,show logging buffered | include self,check scan channel list.result 1.  3) wifi0 dual mode.  4) wait 2 minutes,show logging buffered | include self,check scan channel list.result 1. | | |
| Expect result | Result 1:36~64,100~116,132~140,149~165. | | |
| Test result |  | | |
| Comment | 1) backhaul mode:  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:9720 Access 1 36 -25 yes 0 0 0 20  0019:7745:4cd0 Access cathy 1 -32 yes 3 0 1 20  0019:7703:89a0 Backhaul hive0 48 -42 yes 0 0 0 40-  0019:7703:8990 Backhaul hive0 11 -24 yes 2 0 0 40-  0019:7745:4ce0 Backhaul hive0 153 -38 yes 0 0 0 20  2) dual mode:  AH-8d0f00#  AH-8d0f00#sh acsp nei  wifi0(14) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7745:9720 Access 1 36 -25 yes 0 0 0 20  0019:7745:4cd0 Access cathy 1 -32 yes 6 0 1 20  0019:7745:4ce0 Backhaul hive0 36 -47 yes 0 0 0 20  0019:7703:89a0 Backhaul hive0 48 -42 yes 0 0 0 40-  0019:7703:8990 Backhaul hive0 11 -27 yes 3 0 0 40- | | |

## Check if mp mesh with correct neighbor when do quick failover through bgscan result.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_24 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use | AP1-----SW-----AP110 | | |
| Description | check if mp mesh with correct neighbor when do quick failover through bgscan result. | | |
| Pre-condition | AP1:  1) Creat a new radio profile ng,phymode 11ng,bind to wifi0.  2) wifi0 backhaul mode,channel 11.  3) Creat a new radio profile na,phymode 11na,bind to wifi1.  4) wifi1 radio channel 36,power 1.  5) creat hive pyy,bind to mgt0.  AP2:  1) Creat a new radio profile na,phymode 11na,bind to wifi0.  2) set radio profile na bgscan interval 1 minute.  3) creat hive pyy,bind to mgt0. | | |
| Test procedure | 1) show acsp \_nbr,check if ap110 find all neighbor.  2) shutdown ap110 eth0.result 1. | | |
| Expect result | Result 1:ap110 should do quick failover through bgscan result mesh with channel 36 although channel 11 has stronger power. | | |
| Test result |  | | |
| Comment | 1) 11na:  AH-0f12c0#  AH-0f12c0#sh int  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:770f:12c0 - U - 1 - pyy -  Eth0 0019:770f:12c0 backhaul U - 1 - pyy -  Wifi0 0019:770f:12d0 backhaul U 40 - na - -  Wifi0.1 0019:770f:12d0 backhaul U 40 1 na pyy -  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 36 29:1846 MA 00010100 STATIC 86 1 0 0 pyy fc6b8dc 0  0019:7740:5d90 0019:7740:5d90 0019:7740:5d80 172.16.91.24 11 50:1 MA 00010080 STATIC 198 20 0 0 pyy fc6b8dc 0  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp nei  wifi0(5) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7740:5da0 Backhaul pyy 36 -66 yes 0 0 0 20  0019:7740:5d90 Backhaul pyy 11 -45 yes 2 0 0 20  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#int e0 sh  eth0: Link down  AH-0f12c0#2000-01-01 00:07:03 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal unreachable, start failover trigger timer(2 seconds).  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: ACSP failover trigger timer(portal unreachable) kick in wifi0(5)  2000-01-01 00:07:05 debug ah\_dcd: wifi0(5): handling portal unreachable event when radio channel is static channel.  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: get available channel : 9  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): change state from disable to run.  2000-01-01 00:07:05 debug ah\_dcd: wifi0(5): start do failover because of portal unreachable.  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: Nbr tbl count 2, Chnl tbl count 1  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: ACSP mesh path(mac: 0019:7740:5da0, chan: 36): cost(20), interf\_cost(0), crc\_cost(0), sta\_cost(0), rssi\_cost(20)(avg rssi(29)), hop\_cost(0).  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: ACSP mesh Found cost(20), avg rssi(29), count(1).  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): quick failover through bg scan result.  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): find channel: 36, cost 20  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN\_DFS state  2000-01-01 00:07:05 debug kernel: Last VAP (wifi0) DOWN: Stop AWE specific timers  2000-01-01 00:07:05 debug kernel: 1st VAP (wifi0) UP: Start AWE specific timers  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN state  2000-01-01 00:07:05 debug ah\_dcd: [dcd\_chnl]: wifi0(5): RUN state  2000-01-01 00:07:14 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal reachable, stop failover trigger timer.  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Enable 36 Enable 20  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp ch  wifi0 (5):  State: RUN  Lowest cost channel: 36, lowest-cost: 20  AP type: mesh  AH-0f12c0#sh hive pyy nei  Chan=channel number; Pow=Power in dBm;  A-Mode=Authentication mode; Cipher=Encryption mode;  Conn-Time=Connected time; Hstate=Hive State;  Mac Addr Chan Tx Rate Rx Rate Pow(SNR) A-Mode Cipher Conn-Time Hstate Phymode Chan-width Hive  -------------- ---- ------- ------- --- ---------- -------- --------- -------- ------- ---------- ----  0019:7740:5da0 36 130M 130M -61(38) open aes ccmp 00:00:27 Auth 11na 20MHz pyy  Total neighbor count: 1  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 36 30:2345 MA 00010100 STATIC 0 1 0 0 pyy fc6b8dc 0  0019:7740:5d90 0019:7740:5d90 0019:7740:5d80 172.16.91.24 11 50:1 MA 00010080 STATIC 266 20 0 0 pyy fc6b8dc 0  AH-0f12c0#  2) 11ng:  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh int  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:770f:12c0 - U - 1 - pyy -  Eth0 0019:770f:12c0 backhaul U - 1 - pyy -  Wifi0 0019:770f:12d0 backhaul U 6 - ng - -  Wifi0.1 0019:770f:12d0 backhaul U 6 1 ng pyy -  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Enable 6 Enable 20  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7740:5d90 0019:7740:5d90 0019:7740:5d80 172.16.91.24 11 32:1 MA 00010080 STATIC 29 1 0 0 pyy fc6b8dc 0  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 36 49:1 MA 00010100 STATIC 28 20 0 0 pyy fc6b8dc 0  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#int e0 sh  eth0: Link down  AH-0f12c0#2000-01-01 02:47:33 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal unreachable, start failover trigger timer(2 seconds).  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: ACSP failover trigger timer(portal unreachable) kick in wifi0(5)  2000-01-01 02:47:34 debug ah\_dcd: wifi0(5): start do failover because of portal unreachable.  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: Nbr tbl count 2, Chnl tbl count 2  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: ACSP mesh path(mac: 0019:7740:5d90, chan: 11): cost(20), interf\_cost(0), crc\_cost(0), sta\_cost(0), rssi\_cost(20)(avg rssi(32)), hop\_cost(0).  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: ACSP mesh Found cost(20), avg rssi(32), count(1).  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: wifi0(5): quick failover through bg scan result.  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: wifi0(5): find channel: 11, cost 20  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN\_DFS state  2000-01-01 02:47:34 debug kernel: Last VAP (wifi0) DOWN: Stop AWE specific timers  2000-01-01 02:47:34 debug kernel: 1st VAP (wifi0) UP: Start AWE specific timers  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN state  2000-01-01 02:47:34 debug ah\_dcd: [dcd\_chnl]: wifi0(5): RUN state  AH-0f12c0#  AH-0f12c0#2000-01-01 02:47:37 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal reachable, stop failover trigger timer.  AH-0f12c0#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Enable 11 Enable 20  AH-0f12c0#  AH-0f12c0#sh hive pyy nei  Chan=channel number; Pow=Power in dBm;  A-Mode=Authentication mode; Cipher=Encryption mode;  Conn-Time=Connected time; Hstate=Hive State;  Mac Addr Chan Tx Rate Rx Rate Pow(SNR) A-Mode Cipher Conn-Time Hstate Phymode Chan-width Hive  -------------- ---- ------- ------- --- ---------- -------- --------- -------- ------- ---------- ----  0019:7740:5d90 11 117M 130M -58(30) open aes ccmp 00:00:09 Auth 11ng 20MHz pyy  Total neighbor count: 1  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh amrp nei  wifi0.1:  0019:7740:5d90 172.16.91.24 TWO-WAY cost mine/peer(1200/1200) lastRx-u/b(0/4 sec) flip(0/0/1200)  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp nei  wifi0(5) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7740:5d90 Backhaul pyy 11 -66 yes 6 0 1 20  0019:7740:5da0 Backhaul pyy 36 -46 yes 0 0 0 20  AH-0f12c0#  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7740:5d90 0019:7740:5d90 0019:7740:5d80 172.16.91.24 11 29:261 MA 00010080 STATIC 0 1 0 0 pyy fc6b8dc 0  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 36 49:1 MA 00010100 STATIC 62 20 0 0 pyy fc6b8dc 0  AH-0f12c0#  AH-0f12c0# | | |

## Disable AP110 dfs channel,check if it will mesh with dfs channel neighbor by quick failover through bgscan result.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_25 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use | AP1----SW----AP2  |  AP110 | | |
| Description | Disable AP110 dfs channel,check if it will mesh with dfs channel neighbor by quick failover through bgscan result. | | |
| Pre-condition | AP1:  1) wifi1 channel 36.  2) wifi1 radio power 1.  3) creat hive pyy,bind to mgt0.  AP2:  1) wifi1 channel 52.  2) creat hive pyy.bind to mgt0.  AP110:  1) creat a radio profile na,phymode 11na,bind to wifi0.  2) creat hive pyy,bind to mgt0.  3) show interface wifi0,make sure no dfs channel. | | |
| Test procedure | 1) show acsp \_nbr,check if ap110 find all neighbor.  2) shutdown ap110 eth0.result 1. | | |
| Expect result | Result 1:ap110 should do quick failover through bgscan result mesh with channel 36 although channel 52 has stronger power. | | |
| Test result |  | | |
| Comment | AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7744:73a0 0019:7744:73a0 0019:7744:7380 172.16.91.23 36 28:112 MA 00020100 STATIC 94 1 0 0 pyy fc6b8dc 0  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 52 52:1 MA 00010300 STATIC 9 20 0 0 pyy fc6b8dc 0  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#\_deb dcd ch  AH-0f12c0#  AH-0f12c0#  debug chnl turned on (0x8)  deb con  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Enable 157 Enable 20  AH-0f12c0#int e0 sh  eth0: Link down  AH-0f12c0#2000-01-01 00:02:28 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal unreachable, start failover trigger timer(2 seconds).  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: ACSP failover trigger timer(portal unreachable) kick in wifi0(5)  2000-01-01 00:02:30 debug ah\_dcd: wifi0(5): start do failover because of portal unreachable.  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: Nbr tbl count 2, Chnl tbl count 2  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: ACSP mesh path(mac: 0019:7744:73a0, chan: 36): cost(20), interf\_cost(0), crc\_cost(0), sta\_cost(0), rssi\_cost(20)(avg rssi(28)), hop\_cost(0).  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: ACSP mesh Found cost(20), avg rssi(28), count(1).  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: wifi0(5): quick failover through bg scan result.  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: wifi0(5): find channel: 36, cost 20  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN\_DFS state  2000-01-01 00:02:30 debug kernel: Last VAP (wifi0) DOWN: Stop AWE specific timers  2000-01-01 00:02:30 debug kernel: 1st VAP (wifi0) UP: Start AWE specific timers  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN state  2000-01-01 00:02:30 debug ah\_dcd: [dcd\_chnl]: wifi0(5): RUN state  2000-01-01 00:02:35 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal reachable, stop failover trigger timer.  AH-0f12c0#  AH-0f12c0#sh hive pyy nei  Chan=channel number; Pow=Power in dBm;  A-Mode=Authentication mode; Cipher=Encryption mode;  Conn-Time=Connected time; Hstate=Hive State;  Mac Addr Chan Tx Rate Rx Rate Pow(SNR) A-Mode Cipher Conn-Time Hstate Phymode Chan-width Hive  -------------- ---- ------- ------- --- ---------- -------- --------- -------- ------- ---------- ----  0019:7744:73a0 36 117M 117M -72(28) open aes ccmp 00:00:10 Auth 11na 20MHz pyy  Total neighbor count: 1  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp nei  wifi0(5) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7744:73a0 Backhaul pyy 36 -67 yes 0 0 1 40+  0019:7740:5da0 Backhaul pyy 52 -43 yes 0 0 0 20  AH-0f12c0#sh amrp nei  wifi0.1:  0019:7744:73a0 172.16.91.23 TWO-WAY cost mine/peer(480/480) lastRx-u/b(0/2 sec) flip(0/0/480)  AH-0f12c0#  AH-0f12c0#sh int  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:770f:12c0 - U - 1 - pyy -  Eth0 0019:770f:12c0 backhaul D - 1 - pyy -  Wifi0 0019:770f:12d0 backhaul U 36 - na - -  Wifi0.1 0019:770f:12d0 backhaul U 36 1 na pyy -  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh int w0 ch  Channel 36 : 5180 Mhz 11na 20 40U Channel 40 : 5200 Mhz 11na 20 40L  Channel 44 : 5220 Mhz 11na 20 40U Channel 48 : 5240 Mhz 11na 20 40L  Channel 149 : 5745 Mhz 11na 20 40U Channel 153 : 5765 Mhz 11na 20 40L  Channel 157 : 5785 Mhz 11na 20 40U Channel 161 : 5805 Mhz 11na 20 40L  Channel 165 : 5825 Mhz 11na 20  AH-0f12c0#  AH-0f12c0#  AH-0f12c0# | | |

## AP110 outdoor mode,check if it mesh with correct neighbor by quick failover through bgscan result.

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Dual\_band\_bgscan\_function\_26 | | |
| Priority | Middle | Automation Flag | N/A |
| Topology to use | AP1----SW----AP2  |  AP110 | | |
| Description | AP110 outdoor mode,check if it mesh with correct neighbor by quick failover through bgscan result. | | |
| Pre-condition | AP1:  1) wifi1 channel 36.  2) creat hive pyy,bind to mgt0.  AP2:  1) wifi1 channel 149.  2) wifi1 radio power 1.  3) creat hive pyy.bind to mgt0.  AP110:  1) creat a radio profile na,phymode 11na,bind to wifi0.  2) creat hive pyy,bind to mgt0. | | |
| Test procedure | 1) show acsp \_nbr,check if ap110 find all neighbor.  2) shutdown ap110 eth0.result 1. | | |
| Expect result | Result 1:ap110 should do quick failover through bgscan result mesh with channel 149 although channel 36 has stronger power. | | |
| Test result |  | | |
| Comment | AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh int  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:770f:12c0 - U - 1 - pyy -  Eth0 0019:770f:12c0 backhaul U - 1 - pyy -  Wifi0 0019:770f:12d0 backhaul U 165 - na - -  Wifi0.1 0019:770f:12d0 backhaul U 165 1 na pyy -  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp  Interface Channel select state Channel Power ctrl state Tx power(dbm)  --------- --------------------- -------- --------------------- -------------  Wifi0 Enable 165 Enable 20  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#sh acsp nei  wifi0(5) ACSP neighbor list:  Bssid Mode Ssid/Hive Chan Rssi(dBm) Aerohive AP CU CRC STA Channel-width  0019:7740:5da0 Backhaul pyy 52 -43 yes 0 0 0 20  0019:7744:73a0 Backhaul pyy 149 -69 yes 0 0 0 40+  AH-0f12c0#sh acsp \_nbr  wifi0(5) ACSP neigbor list:  MACADDR RADIO\_ID NODE\_ID NODE\_IP CHAN S:NS FLAG CH\_FLAG STATE AGE POW MBO DBO SSID/MESHID HVPW\_HASH METRIC  0019:7740:5da0 0019:7740:5da0 0019:7740:5d80 172.16.91.24 52 52:32 MA 00010300 STATIC 54 20 0 0 pyy fc6b8dc 0  0019:7744:73a0 0019:7744:73a0 0019:7744:7380 172.16.91.23 149 26:29 MA 00020100 STATIC 51 1 0 0 pyy fc6b8dc 0  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#int e0 sh  eth0: Link down  AH-0f12c0#2000-01-01 00:01:48 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal unreachable, start failover trigger timer(2 seconds).  2000-01-01 00:01 Scan in progress.. Cancelling it  :49 debug ah\_dcd: [dcd\_chnl]: ACSP failover trigger timer(portal unreachable) kick in wifi0(5)  2000-01-01 00:01:49 debug ah\_dcd: wifi0(5): start do failover because of portal unreachable.  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: Nbr tbl count 2, Chnl tbl count 2  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: ACSP mesh path(mac: 0019:7744:73a0, chan: 149): cost(20), interf\_cost(0), crc\_cost(0), sta\_cost(0), rssi\_cost(20)(avg rssi(26)), hop\_cost(0).  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: ACSP mesh Found cost(20), avg rssi(26), count(1).  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: wifi0(5): quick failover through bg scan result.  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: wifi0(5): find channel: 149, cost 20  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN\_DFS state  2000-01-01 00:01:49 debug kernel: Last VAP (wifi0) DOWN: Stop AWE specific timers  2000-01-01 00:01:49 debug kernel: 1st VAP (wifi0) UP: Start AWE specific timers  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: wifi0(5): LISTEN state  2000-01-01 00:01:49 debug ah\_dcd: [dcd\_chnl]: wifi0(5): RUN state  AH-0f12c0#sh hive pyy nei  Chan=channel number; Pow=Power in dBm;  A-Mode=Authentication mode; Cipher=Encryption mode;  Conn-Time=Connected time; Hstate=Hive State;  Mac Addr Chan Tx Rate Rx Rate Pow(SNR) A-Mode Cipher Conn-Time Hstate Phymode Chan-width Hive  -------------- ---- ------- ------- --- ---------- -------- --------- -------- ------- ---------- ----  0019:7744:73a0 149 6M 6M -67(27) open aes ccmp 00:00:09 Auth 11na 20MHz pyy  Total neighbor count: 1  AH-0f12c0#  2000-01-01 00:02:02 debug ah\_dcd: [dcd\_chnl]: wifi0(5): portal reachable, stop failover trigger timer.  AH-0f12c0#sh amrp nei  wifi0.1:  0019:7744:73a0 172.16.91.23 TWO-WAY cost mine/peer(480/480) lastRx-u/b(2/8 sec) flip(0/0/480)  AH-0f12c0#  AH-0f12c0#  AH-0f12c0#  AH-0f12c0# | | |

# CLI (Automation Status: Yes/No)

# <Just list all cli that this feature has one by one>

# Customer Issue or Typical Bug