

Band Steering User manual(With CcspWifiAgent) - RDKB

- [Introduction](#)
- [Environment Setup](#)
- [Executing System](#)
- [Troubleshooting](#)
 - [Error Messages](#)
 - [Special Considerations](#)
- [Support](#)

Introduction

Band Steering is a solution ensures that clients are connected to the best radio. Dual Band supported Gateway can transmit SSIDs in both 2.4GHz and 5GHz frequency band. Enabling and disabling of Band Steering can be done through dmcli command line utility.

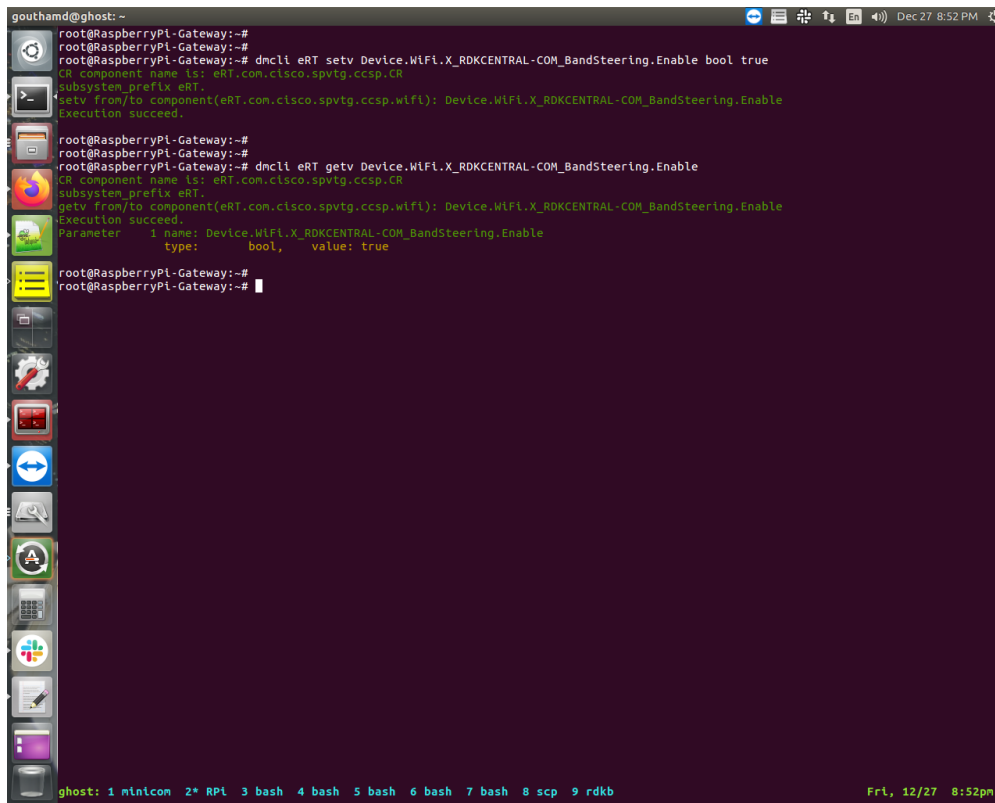
Environment Setup

Band Steering feature requires dual band capability i.e device should be able to broadcast SSID in both 2.4GHz and 5GHz frequency. Also this feature will be of use and applicable only to 5GHz capable device. Older 2.4 GHz-only capable devices cannot benefit from this feature. Still older 2.4GHz-only clients can connect and operate on 2.4GHz frequency only.

Executing System

Band Steering by default will be disabled. The user can enabled it using dmcli command as shown below.

Enabling Band Steering Example:



```
gouthamd@ghost:~$ root@RaspberryPi-Gateway:~#
root@RaspberryPi-Gateway:~# dmcli eRT setv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable bool true
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
setv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.

root@RaspberryPi-Gateway:~#
root@RaspberryPi-Gateway:~# dmcli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.
Parameter      1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
                type:      bool,      value: true

root@RaspberryPi-Gateway:~#
root@RaspberryPi-Gateway:~#
```

After enabling of Band Steering, the SSID broadcasted by the two interfaces are same. A 5GHz capable client will normally connect to a 5GHz client and older 2.4GHz client will connect to 2.4GHz frequency band.

Both Bands having same SSID:

```
gouthamd@ghost:~
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# dncli eRT setv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable bool true
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
setv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.

root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# dncli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
type: bool, value: true

root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# iw dev
phy#2
    Interface wlan2
        ifindex 5
        wdev 0x200000001
        addr b8:27:eb:61:1d:aa
        ssid RPi3_RDKB-AP0
        type AP
        channel 6 (2437 MHz), width: 20 MHz, center1: 2437 MHz
        txpower 31.00 dBm
phy#1
    Interface wlan1
        ifindex 4
        wdev 0x100000001
        addr 1a:d6:c7:08:c6:c1
        type managed
        txpower 12.00 dBm
phy#0
    Interface wlan0
        ifindex 3
        wdev 0x1
        addr 18:d6:c7:08:c6:c1
        ssid RPi3_RDKB-AP0
        type AP
        txpower 12.00 dBm
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
ghost: 1 minicom 2* RPi 3 bash 4 bash 5 bash 6 bash 7 bash 8 scp 9 rdkb
Fri, 12/27 8:54pm
```

Disabling of Band Steering will reset all previously made configuration on the 5GHz band. Band Steering can be disabled by the below command.

Disabling Band Steering Example:

```
gouthamd@ghost:~
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# dncli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
type: bool, value: true

root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# dncli eRT setv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable bool false
CR component name is: eRT.com.cisco.spvtg.ccsp.CR
subsystem_prefix eRT.
setv from/to component(eRT.com.cisco.spvtg.ccsp.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.Enable
Execution succeed.

root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~#
root@RasperryPI-Gateway:~# iw dev
phy#2
    Interface wlan2
        ifindex 5
        wdev 0x200000001
        addr b8:27:eb:61:1d:aa
        ssid RPi3_RDKB-AP0
        type AP
        channel 6 (2437 MHz), width: 20 MHz, center1: 2437 MHz
        txpower 31.00 dBm
phy#1
    Interface wlan1
        ifindex 4
        wdev 0x100000001
        addr 1a:d6:c7:08:c6:c1
        type managed
        txpower 12.00 dBm
phy#0
    Interface wlan0
        ifindex 3
        wdev 0x1
        addr 18:d6:c7:08:c6:c1
        ssid RPi3_RDKB-AP1
        type AP
        txpower 12.00 dBm
root@RasperryPI-Gateway:~#
ghost: 1 minicom 2* RPi 3 bash 4 bash 5 bash 6 bash 7 bash 8 scp 9 rdkb
Fri, 12/27 9:03pm
```

Setting the Threshold value Example:

```

root@RaspberryPi-Gateway:~# dmcli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.UtilizationThreshold
           type:  int,    value: 0
Parameter 2 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.RSSIThreshold
           type:  int,    value: -100
Parameter 3 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.PhyRateThreshold
           type:  int,    value: 0
Parameter 4 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.OverloadInactiveTime
           type:  int,    value: 0
Parameter 5 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.IdleInactiveTime
           type:  int,    value: 0

root@RaspberryPi-Gateway:~# dmcli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.UtilizationThreshold
           type:  int,    value: 0
Parameter 2 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.RSSIThreshold
           type:  int,    value: -100
Parameter 3 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.PhyRateThreshold
           type:  int,    value: 0
Parameter 4 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.OverloadInactiveTime
           type:  int,    value: 0
Parameter 5 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.IdleInactiveTime
           type:  int,    value: 0

root@RaspberryPi-Gateway:~# dmcli eRT setv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.RSSIThreshold int -25
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
setv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.RSSIThreshold
Execution succeed.

root@RaspberryPi-Gateway:~# dmcli eRT setv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.RSSIThreshold int -25
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
setv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.RSSIThreshold
Execution succeed.

```

```

root@RaspberryPi-Gateway:~# dmcli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.UtilizationThreshold
           type:  int,    value: 0
Parameter 2 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.RSSIThreshold
           type:  int,    value: -25
Parameter 3 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.PhyRateThreshold
           type:  int,    value: 0
Parameter 4 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.OverloadInactiveTime
           type:  int,    value: 0
Parameter 5 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.1.IdleInactiveTime
           type:  int,    value: 0

root@RaspberryPi-Gateway:~# dmcli eRT getv Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.
CR component name is: eRT.com.cisco.spvtg.ccsd.CR
subsystem_prefix eRT.
getv from/to component(eRT.com.cisco.spvtg.ccsd.wifi): Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.
Execution succeed.
Parameter 1 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.UtilizationThreshold
           type:  int,    value: 0
Parameter 2 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.RSSIThreshold
           type:  int,    value: -25
Parameter 3 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.PhyRateThreshold
           type:  int,    value: 0
Parameter 4 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.OverloadInactiveTime
           type:  int,    value: 0
Parameter 5 name: Device.WiFi.X_RDKCENTRAL-COM_BandSteering.BandSetting.2.IdleInactiveTime
           type:  int,    value: 0

```

Client connected to 5GHz initially Example:

```

root@RaspberryPi-Gateway:~# iw dev wlan0 station dump
Station <Mac Address> (on wlan0)
    inactive time: 3520 ms
    signal:       -21 dBm

```

Client after switching to 2.4GHz when RSSI signal goes above -25dBm Example:

```

root@RaspberryPi-Gateway:~# iw dev wlan2 station dump
Station <Mac Address> (on wlan2)
    inactive time: 6000 ms
    rx bytes: 39519
    rx packets: 280
    tx bytes: 58423
    tx packets: 267
    tx failed: 0
    signal: -50 [-50] dBm
    tx bitrate: 72.2 MBit/s
    rx bitrate: 1.0 MBit/s
    authorized: yes
    authenticated: yes
    associated: yes
    WMM/WME: yes
    TDLS peer: yes
    DTIM period: 2
    beacon interval: 100
    short slot time: yes
    connected time: 197 seconds

```

Client association and disassociation event capture:

```

root@RaspberryPi-Gateway:~# iw event -f
wlan0: del station <Mac addr>
wlan0: del station <Mac addr>
wlan0: del station <Mac addr>
wlan1: new station <Mac addr>
wlan1: del station <Mac addr>
wlan0: new station <Mac addr>
wlan1: new station <Mac addr>
wlan0: new station <Mac addr>

```

Troubleshooting

- **Error Messages**

If enabling of the Band Steering fails, then both the band will have different SSIDs, still Wifi would operate and wireless clients can connect to them.

- **Special Considerations**

Device should be dual band capable to support Band Steering functionality.

Support

Contact	Organization	Phone	Email	Role
Rajkumar Narayanan	L&T Technology Service Ltd.	NIL	rajkumar.narayanan@lts.com	Program manager

