Aerohive Networks Inc.

BRxxx Ethernet Test Cases

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 0.1 | 12/07/2011 | wanwan | Initial version |
| 0.2 | 12/15/2011 | wanwan | Add detail steps for each cases |
| 0.3 | 12/21/2011 | wanwan | Update test cases based on case review and zhang ligang’s suggestions |
| 0.4 | 12/27/2011 | wanwan | Add upgrade HiveOS test case and CLI test casesiHi |
| 0.5 | 9/11/2013 | Wanwan | Add a new case for dhcp of a bug: BR\_LanPorts\_datapath\_9 |

Table of Contents

Glossary and Abbreviations

# Introduction

BRxxx has one wan port and 4 switched lan ports. 4 swithed lan ports are in the eye of sw modules (AMRP/FE/AUTH) in Casablanca. This means all traffics will be sent to CPU. But locally forwarding packets among lan ports should be handled by HW

For the lan ports, we need to provide vlan (port based vlan / 802.1q trunk vlan) support with authentication capabilities (cwp, 802.1x and mac acl).

The wan port and lan ports function block is depicted as below:

CPU

CPU

GMAC1

GMAC0

MAC

L2

**Reth1**

Virtual mac1 vmac2 vmac3 vmac4

Eth0

Eth4

Eth1

Eth2

Eth3

PHY

L1

WAN PHY1 PHY2 PHY3 PHY4

For BRxxx, from cpu point, it just has 2 physical Ethernet interfaces:

Eth0 (wan port) associated with GMAC0. [This document we don’t focus on wan port (eth0)]

Reth1 associated with GMAC1, based on Reth1, there is 4 interfaces (eth1-eth4) mapped to lan port1-port4, and the 4 interfaces has four virtual mac address for them.

**Processing procedure on lan ports:**

**Port Based mode:**

* **For ingress:**

When a client plug-in, hw will update mac-table, put client’s mac and associated port into mac-table.

**Port process procedure:**

If received packet from port1 to port4, sw driver get the src-mac from packet, lookup mac-table, retrieve port id, and then assign the port id to skb->dev field.

**Vlan process procedure:**

Port based lan ports have a native vlan.

If received packet from port 1 to port 4, once driver learnt port ID, sw driver will lookup vlan-table, retrieve vlan id, then assign it to skb->vlan\_tci field.

* **For egress:**

For BR100 and BR200, vlan process is the same. After fe process is done, sw driver can get vlan tag from skb->vlan\_tci.

**BR100:**

Get packet forwarding port from skb->dev (ethx), then hw will decide to send out the packet based on vlan and HW ARL table(mac-table) mapped forwarding ports.

**BR200:**

Get packet forwarding port from skb-dev(eth1.x), sw will create a Atheros Header for per packet. It identify the port id, hw will use this port to transmit the packet directly.

[Since BR200 chip support Atheros Header, when create it, set from\_cpu/dp\_bit\_mp fields, hw don’t need to lookup hw ARL table (mac-table)]

**Trunk mode (802.1q):**

* **For ingress:**

**Port process procedure:**

For BR100, when hw received packets, sw driver assign the ethx to skb->dev.

For BR200, when received packets, sw driver assign the trunk ports to skb->dev.

**Vlan process procedure:**

Packets received from trunk port can have either vlan tag or no tag.

If has no tag, sw driver will assign the native vlan ID for this packets, and put the vlan tag to skb->vlan\_tci field.

If has vlan tag, and the vlan tag maches the allowed vlan list, sw driver need to retrieve the vlan id and put into skb->vlan\_tci field.

**Support vlan-id-check.**

* **For egress:**

When packets transmitted on a trunk port, should have skb->vlan\_tci and skb->dev fields.

**Vlan process procedure:**

sw driver will check skb->vlan\_tci with native vlan id:

if it matches, no vlan tag will be added. Otherwise the vlan tag will be added.

**Port process procedure:**

For BR100, transmit the packets by ethx, vlan-table and mac-table.

For BR200, transmit the packets directly with the trunk port bit (Atheros Header dp\_bit\_map) bypassing HW ARL table lookup.

# Test Objectives

Four lan ports is visible for sw as virtual interfaces (eth1/eth2/eth3/eth4)

1. Lan to lan local packets forwarding support
2. lan to wan/wifi

The below cases should be covered

* basic feature: mac-table, mac learning
* Vlan test for BR100/BR200
* ingress/egress on BR100/BR200
* Unicast packets process procedure on BR100/BR200
* Broadcast packets process procedure on BR100/BR200
* Dhcp packets process procedure on BR100/BR200
* Raw Sockets process procedure on BR100/BR200
* Lan ports application, like upid, qos, cwp etc.
* Cli test
* performace

# Test Acceptance Criterion from Development

* Approved – MRD

The link to MRD

* Approved – Functional Specifications

The link to function spec

* Approved – Unit Test Plans

The link to unit test report of dev

# Product Pass Criterion

Meet all objects in marketing requirement or function spec which may include key function objectives, capacity objectives, and performance objectives and so on.

# Test Bed/Topo Design

<List topo and topo ID>

# Test Case

## Key Scenarios

## Function Test Case

### Lan ports feature test

#### Lan ports basic function

##### Lan ports link state check

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_LinkState\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch1  |  BR(eth1-eth4)－laptop1  (eth3)  |  Switch2  |  PC1 | | |
| Description | Check lan ports link state | | |
| Pre-condition | BR connect to switch1  Laptop1 connect to BR lan port eth1/eth2/eth3/eth4  Switch2 connect to BR lan port eth3  PC1 connect to switch2 | | |
| Test procedure | ***BR connect a laptop:***   1. Plug –in laptop1 to BR lan ports eth1 2. －check link state of interface eth1   ***#show interface***  ***#show lan port-config***  －check duplex, speed of interface eth1  ***#show interface eth1***  ***Result1***   1. Unplug laptop1, then plug-in laptop1 to BR lan ports eth2 2. －check link state of BR interface eth1/eth2   ***#show interface***  ***#show lan port-config***  －check duplex, speed of interface eth2  ***#show interface eth2***  ***Result2***   1. Unplug laptop1, then plug-in laptop1 to BR lan ports eth3 2. －check link state of BR interface eth2/eth3   ***#show interface***  ***Show lan port-config***  －check duplex, speed of interface eth3  ***#show interface eth3***  ***Result3***   1. Unplug laptop1, then plug-in laptop1 to BR lan ports eth4 2. check link state of BR interface eth3/eth4   ***#show interface***  ***#show lan port-config***  －check duplex, speed of interface eth4  ***#show interface eth4***  ***Result4***   1. unplug cable switch2 connected to BR, then plug into another lan port, such as lan port eth3   －check link state of BR lan port eth3  ***#show interface***  ***#show lan port-config***  －check duplex, speed of interface eth3  ***#show interface eth3***  ***Result5***  ***BR connect a switch to lan port ethx***  ***Such as switch2 connect to BR lan port eth3:***   1. PC1 connect to switch2 port1, then move to switch2 port2, 2. check link state of BR interface eth3 ***Result6***   ***#show interface***  ***#show lan port-config*** | | |
| Expect result | ***Result1:*** interface eth1 is up, duplex is full-duplex and speed is 100Mbps  Laptop1 can obtain ip and access internet  ***Result2:***  interface eth1 is down, eth2 is up, duplex is full-duplex and speed is 100Mbps  Laptop1 can obtain ip and access internet  ***Result3:*** interface eth2 is down, eth3 is up, duplex is full-duplex and speed is 100Mbps  Laptop1 can obtain ip and access internet  ***Result4:***  interface eth3 is down, eth4 is up, duplex is full-duplex and speed is 100Mbps  Laptop1 can obtain ip and access internet  ***Result5:*** interface eth3 is up, the duplex is full-duplex and speed is 100Mbps  PC1 which connects to switch2 can obtain ip and access internet  ***Result6:***  when PC1 moved from switch2 port1 to switch2 port2, interface eth3 is always up  PC1 can obtain ip and access internet. | | |
| Test result | BR100 Result:  ***pass***  BR200 Result: | | |
| Comment |  | | |

##### HW ARL Table Support (plug/unplug laptop)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_ARLTable\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch1  |  (eth0)BR(eth1-eth4)－laptop1  (eth3)  |  Switch2  |  PC1 | | |
| Description | Check mac-table entry when plug/unplug laptop1 | | |
| Pre-condition | BR connect to switch1  Laptop1 connect to BR lan port eth1/eth2/eth3/eth4  Switch2 connect to BR lan port eth3  PC1 connect to switch2 | | |
| Test procedure | 1. Plug in laptop1 to lan port eth1. Check mac-table. ***Result1***   ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. Plug in laptop1 to lan port eth2. Check mac-table. ***Result2***   ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. Plug in laptop1 to lan port eth3. Check mac-table. ***Result3***   ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. Plug in laptop1 to lan port eth4. Check mac-table. ***Result4***   ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. Unplug laptop1. Check mac-table. ***Result5***   ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. *-Plug switch2 to lan port eth3.*   *-Plug PC1 to switch2.*  Check mac-table  ***Result6***  ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]*** | | |
| Expect result | ***Result1:***  **-**mac-table insert an entry for eth1, port field is lan port eth1, mac address field is laptop1’s mac  **-**mac-learning table has an entry for eth1  ***Result2:***  ***-*** mac-table insert an entry for eth2, port field is lan port eth2, mac address field is laptop1’s mac  **-**mac-learning table has an entry for eth2  ***Result3:***  **-**mac-table insert an entry for eth3, port field is lan port eth3, mac address field is laptop1’s mac  **-**mac-learning table has an entry for eth3  ***Result4:***  ***-*** mac-table insert an entry for eth4, port field is lan port eth4, mac address field is laptop1’s mac  **-**mac-learning table has an entry for eth4  ***Result5:***  the entry of mac-table/mac-learning table for laptop1 is removed immediately  ***Result6:*** mac-table/mac-learning table has entries for switch2’s ip and PC1’s ip | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment | BR100: bug 15980 fixed now: as described in the bug, we cannot be 100% real time, worse delay is 10s, average is 5s | | |

##### Mac-table support (lan port shutdown)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_ARLTable\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch1  |  (eth0)BR(eth1-eth4)－laptop1  (eth3)  |  Switch2  |  PC1 | | |
| Description | Check mac-table entry when shutdown lan ports | | |
| Pre-condition | BR connect to switch1  Laptop1 connect to BR lan port eth1/eth2/eth3/eth4  Switch2 connect to BR lan port eth3  PC1 connect to switch2 | | |
| Test procedure | 1. Laptop1 connect to lan port eth1. 2. ***Shutdown lan port eth1*** by cli, then check mac-table and laptop1 status ***Result1***   ***#interface eth1 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. ***No shutdown lan port eth1*** by cli, then check mac-table, and laptop1 status ***Result2***   ***#no interface eth1 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth1 mac-learning all [SW]***   1. Laptop1 connect to lan port eth2. 2. ***Shutdown lan port eth2*** by cli, then check mac-table. ***Result3***   ***#interface eth2 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth2 mac-learning all [SW]***   1. ***No shutdown lan port eth2*** by cli, then check mac-table, and laptop1 status ***Result4***   ***#no interface eth2 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth2 mac-learning all [SW]***   1. Laptop1 connect to lan port eth3. 2. ***Shutdown lan port eth3*** by cli, then check mac-table. ***Result5***   ***#interface eth3 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth3 mac-learning all [SW]***   1. ***No shutdown lan port eth3*** by cli, then check mac-table, and laptop1 status ***Result6***   ***#no interface eth3 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth3 mac-learning all [SW]***   1. Laptop1 connect to lan port eth4. 2. ***Shutdown lan port eth4*** by cli, then check mac-table. ***Result7***   ***#interface eth4 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth4 mac-learning all [SW]***   1. ***No shutdown lan port eth4*** by cli, then check mac-table, and laptop1 status ***Result8***   ***#no interface eth4 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth4 mac-learning all [SW]***   1. Switch2 connect to BR lan port eth3, PC1 connect to switch2 2. ***Shutdown lan port eth3*** by cli, then check mac-table. ***Result9***   ***#interface eth3 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth4 mac-learning all [SW]***   1. ***No shutdown lan port eth3*** by cli, then check mac-table, and PC1 status ***Result10***   ***#no interface eth3 shutdown***  ***#show lan mac-table [HW]***  ***#show interface eth4 mac-learning all [SW]*** | | |
| Expect result | ***Result1:***  shutdown lan port eth1, the entry for eth1 is removed immediately, the interface disabled.  ***Result2:***  laptop1 will retrieve ip address successfully, and mac-table will update  ***Result3:***  shutdown lan port eth2, the entry for eth2 is removed immediately, the interface disabled.  ***Result4:*** laptop1 will retrieve ip address successfully, and mac-table will update  ***Result5:***  shutdown lan port eth3, the entry for eth3 is removed immediately, the interface disabled.  ***Result6:*** laptop1 will retrieve ip address successfully, and mac-table will update  ***Result7:***  shutdown lan port eth4, the entry for eth4 is removed immediately, the interface disabled.  ***Result8:*** laptop1 will retrieve ip address successfully, and mac-table will update  ***Result9:***  shutdown lan port eth3, the entry for switch2 and PC1 is removed immediately, the interface disabled.  ***Result10:*** PC1 will retrieve ip address successfully, and mac-table will update | | |
| Test result | BR100: pass  BR200: pass, [lan mac-table/ mac-learning table will removed after age out 5mins] | | |
| Comment |  | | |

##### Mac-table support (maximum entries)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_ARLTable\_4 | | |
| Priority | Middle | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  laptop1 laptop2 | | |
| Description | Mac-table supports maximum of 1k mac, when reach the maximum count, Br will flash mac-table to learn new mac again | | |
| Pre-condition | Laptop1 connect to BR lan port eth1  Laptop2 connect to BR lan port eth4  Lan port eth1 and eth4 with same vlan | | |
| Test procedure | 1. Lan port eth1/eth4 with default vlan 1, mgt0 vlan is 1, dhcp server enable 2. Connect laptop1 to lan port eth1, check laptop1 ip address ***Result1*** 3. On laptop1, use tool HController to send packets with src-mac increasing, check mac-table ***Result2***   ***#show lan mac-table***  ***#show interface eth1 mac-learning all***   1. Then laptop2 connect to lan port eth4, check laptop2 ip address ***Result3***      1. Check mac-table ***Result4***   ***#show lan mac-table***  ***#show interface eth4 mac-learning all*** | | |
| Expect result | ***Result1:*** laptop1 obtain ip address from mgt0 dhcp server successfully  ***Result2:*** check mac-table, there are 1024 entries in mac-table  ***Result3:*** laptop2 obtain ip address from mgt0 dhcp server successfully  ***Result4:*** BR flash mac-table, learnt laptop2’s mac-address. Add this entry for eth4 to mac-table | | |
| Test result | BR100: pass, support 1024 entries. [Failed, connect laptop2 to lan port, can’t get ip address.]  BR200: pass, support 2048 entries. | | |
| Comment |  | | |

##### Lan ports mac-learning (connect a laptop)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_MacLearn\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch1  |  (eth0)BR(eth1-eth4)－laptop1  (eth3)  |  Switch2  |  PC1 | | |
| Description | Check mac learning for lan ports eth1/eth2/eth3/eth4 connect a switch | | |
| Pre-condition | BR connect to switch1  Laptop1 connect to BR lan port eth1/eth2/eth3/eth4  Switch2 connect to BR lan port eth3  PC1 connect to switch2  Mac-learning feature is enabled for lan ports | | |
| Test procedure | 1. Latptop1 connect to lan ports eth1 2. Check route table. ***Result1*** 3. Laptop1 connect to lan ports eth2 4. Check route table. ***Result2*** 5. Laptop1 connect to lan ports eth3 6. Check route table. ***Result3*** 7. Laptop1 connect to lan ports eth4 8. Check route table. ***Result4*** 9. Switch2 connect to BR lan ports eth3   PC1 connect to Switch2.   1. Check route table ***Result5*** | | |
| Expect result | ***Result1:***  show route table, have an entry in table, check laptop1 mac address, and the interface (eth1).  #cli: show route  0015:c5c3:5f21 0019:777a:30c0 eth1 0 0 L  ***Result2:*** check route table, have an entry in table, check laptop1 mac address, and the interface(eth2).  #cli: show route  0015:c5c3:5f21 0019:777a:30c0 eth2 0 0 L  ***Result3:***  check route table, have an entry in table, check laptop1 mac address, and the interface(eth3)  #cli: show route  0015:c5c3:5f21 0019:777a:30c0 eth3 0 0 L  ***Result4:***  check route table, have an entry in table, check laptop1 mac address, and the interface(eth4)  #cli: show route  0015:c5c3:5f21 0019:777a:30c0 eth4 0 0 L  ***Result5:*** check route table, have entries for swith2 and PC1 mac address | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Lan ports mac-learning (connect an AP/BR)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_MacLearn\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)－Laptop1  |  AP  |  Laptop2 | | |
| Description | Check mac learning for lan ports eth1/eth2/eth3/eth4 connect an AP | | |
| Pre-condition | Laptop1 connect to BR  Laptop2 connect to AP  AP connect to BR lan ports ethx  Mac-learning feature is enabled for lan ports in BR  Mac-learning feature is enabled in AP | | |
| Test procedure | 1. Construct the topology 2. Check route table in BR. ***Result1*** 3. Unplug laptop1 from BR, and then connect to AP, check route table in BR and AP. ***Result2*** 4. Unplug AP from BR lan port eth3, and then plug in to another lan port eth4, check route table in BR. ***Result3*** 5. Laptop2 move to BR lan ports Result4 | | |
| Expect result | ***Result1:*** there are 3 entries in route table for laptop1 and AP(mgt0), the interface (oif) is different  ***# show route***  ***#show lan mac-table***  ***Result2:***   * There are entries in route table for laptop1 in AP   ***# show route***  ***#show lan mac-table***   * There are 3 entries in route table for laptop1 and AP(mgt0), but the interface(oif) is same   ***# show route***  ***#show lan mac-table***  ***Result3:*** the mac-table will be updated also. There are 2 entries of lan port eth4 in route table for laptop1 and Ap’s mgt0  ***# show route***  ***#show lan mac-table***  ***Result4:*** mac-table updated immediately, route table updated immediately also  ***# show route***  ***#show lan mac-table*** | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment | bug16241 for br200, it’s by design, after age out, it’s ok | | |

##### Lan ports user profile attribute support for access

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_FT\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | Check user-profile-attribute for lan ports ethx with port-based mode | | |
| Pre-condition | Laptop1 connect to BR lan ports  Lan ports work with port-based mode | | |
| Test procedure | 1. Configure user-profile-attribute for lan ports eth1   ***#interface eth1 mode bridge-access user-profile-attribute 1000***   1. Check lan ports eth1 upid. ***Result1*** 2. Configure user-profile-attribute for lan ports eth2   ***#interface eth2 mode bridge-access user-profile-attribute 1001***   1. Check lan ports eth2 upid. ***Result2*** 2. Configure user-profile-attribute for lan ports eth3   ***#interface eth3 mode bridge-access user-profile-attribute 1002***   1. Check lan ports eth3 upid. ***Result3*** 2. Configure user-profile-attribute for lan ports eth4   ***#interface eth4 mode bridge-access user-profile-attribute 1003***   1. Check lan ports eth4 upid. ***Result4*** | | |
| Expect result | ***Result1:*** the lan port eth1 upid is correct, it’s 1000  # show route  **0019:7744:74c0**  0019:777a:30c0 eth1 0 **1000** L  ***Result2:*** the lan port eth2 upid is correct, it’s 1001  # show route  **0019:7744:74c0**  0019:777a:30c0 eth2 0 **1001** L  ***Result3:*** the lan port eth3 upid is correct, it’s 1002  # show route  **0019:7744:74c0**  0019:777a:30c0 eth3 0 **1002** L  ***Result4:*** the lan port eth4 upid is correct, it’s 1003  # show route  **0019:7744:74c0**  0019:777a:30c0 eth4 0 **1003** L | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Lan ports user profile attribute support for trunk

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_FT\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | Check user-profile-attribute for lan ports ethx with trunk mode | | |
| Pre-condition | Laptop1 connect to BR lan ports  Lan ports work with trunk mode | | |
| Test procedure | 1. Configure user-profile-attribute for lan ports eth1   ***#interface eth1 mode bridge-802.1q user-profile-attribute 1000***   1. Check lan ports lan ports eth1 upid. ***Result1*** 2. Configure user-profile-attribute for lan ports eth2   ***#interface eth2 mode bridge-802.1q user-profile-attribute 1001***   1. Check lan ports lan ports eth2 upid. ***Result2*** 2. Configure user-profile-attribute for lan ports eth3   ***#interface eth3 mode bridge-802.1q user-profile-attribute 1002***   1. Check lan ports lan ports eth3 upid. ***Result3*** 2. Configure user-profile-attribute for lan ports eth4   ***#interface eth4 mode bridge-802.1q user-profile-attribute 1003***   1. Check lan ports lan ports eth4 upid. ***Result4*** | | |
| Expect result | ***Result1:*** the lan port eth1 upid is correct, it’s 1000  # show route  **0019:7744:74c0**  0019:777a:30c0 eth1 0 **1000** L  ***Result2:*** the lan port eth2 upid is correct, it’s 1001  # show route  **0019:7744:74c0**  0019:777a:30c0 eth2 0 **1001** L  ***Result3:*** the lan port eth3 upid is correct, it’s 1002  # show route  **0019:7744:74c0**  0019:777a:30c0 eth3 0 **1002** L  ***Result4:*** the lan port eth4 upid is correct, it’s 1003  # show route  **0019:7744:74c0**  0019:777a:30c0 eth4 0 **1003** L | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Lan ports QoS support

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_FT\_3 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)-----AP  | |  Laptop1 laptop2 | | |
| Description | Check if QoS works for lan ports | | |
| Pre-condition | Laptop1 connect to BR lan ports eth1  Laptop2 connect to BR lan ports eth4  Lan ports are in same vlan  Lan ports mode is port-based/trunk mode  QoS classify selected DSCP | | |
| Test procedure | 1. Configure QoS for lan ports eth1, eth4   eth1:  qos classifier-profile test  interface eth1 qos-classifier test  qos classifier-profile test diffserv  qos classifier-map diffserv 0 5  eth4:  qos marker-profile test  interface eth4 qos-marker test  qos marker-profile test diffserv  qos marker-map diffserv 5 **2**   1. Laptop1 send a packet to laptop2. Capture packet in laptop2, check DSCP field value. ***Result1*** | | |
| Expect result | ***Result1:***  laptop2 can receive the packet.  DSCP value for the captured packet in laptop2 is **2** | | |
| Test result | BR100: **Lan to lan not support QoS**  BR200: **lan to lan not support QoS** | | |
| Comment |  | | |

##### Lan ports CWP support

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_FT\_4 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | Check if lan ports cwp works | | |
| Pre-condition | Laptop1 connect to BR lan ports  Lan ports enable cwp authentication | | |
| Test procedure | 1. Configure passthrough/ecwp for lan ports 2. Enable auth/register authentication 3. Laptop1 connect to lan ports, check if laptop1 can access webpage ***Result1*** 4. then check laptop1 ip address. ***Result2*** 5. Laptop1 request a webpage. ***Result3*** 6. User enters a valid username/password, or register info. ***Result4*** | | |
| Expect result | ***Result1:***  laptop1 can’t access webpage  ***Result2:*** laptop1 get ip from dhcp server successfully  ***Result3:*** laptop1 get a login page.  ***Result4:*** laptop1 pass cwp authentication, and access internet successfully | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

#### Port based mode vlan

##### Lan ports default vlan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_PortBased\_Vlan\_1 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  (eth3)  |  laptop1 | | |
| Description | Lan port works with default config(port based mode), check the lan ports default vlan | | |
| Pre-condition | BR lan port works with default config(port based mode)  Laptop1 connect to lan port eth3 | | |
| Test procedure | 1. Enable mgt0 dhcp/dns server 2. Lan port works with default configure (port based mode) 3. check lan ports mode, default vlan ID.  ***Result1*** 4. Laptop1 connect to lan ports, such as eth3, check HW ARL table (mac-table). ***Result2*** 5. Check laptop ip . ***Result3*** 6. Configure mgt0 native-vlan as 10   *#interface mgt0 native-vlan 10*   1. check if lan ports native-vlan is mapped from mgt0’s native-vlan. ***Result4*** 2. configure user-profile-attribuite for lan ports, such as eth3   #user-profile test10 vlan 10 attribute 10  #interface eth3 mode bridge-access user-profile-attribute 10   1. check lan ports native-vlan ***Result5*** 2. set mgt0.1 vlan as 10, enable mgt0.1 dhcp/dns server   set mgt0.1 ip address 10.10.10.1  set mgt0.1 ip-pool is 10.10.10.10-10.10.10.99   1. laptop1 connect lan ports, such as eth3 2. check laptop1 ip ***Result6*** | | |
| Expect result | ***Result1:*** the default lan ports mode is port based, and default vlan is 1  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***Result2:*** mac-table has an entry for laptop1, indicates port id, laptop mac address  ***#show interface mac-table***  ***Result3:***  laptop1 get an ip address successfully, which is assigned by mgt0’s dhcp server  Laptop1’s ip address is 192.168.85.10  ***Result4:*** lan ports native-vlan should not be mapped from mgt0 native-vlan, the native-vlan is still 1.  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***Result5:*** lan ports native-vlan is mapped from user-profile-attribute  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***#show route***  ***Result6:***  laptop1 get ip address for mgt0.1 dhcp server, laptop1 ip is 10.10.10.10 | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Packets forwarding with/without vlan tag (LAN to LAN)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_PortBased\_Vlan\_2 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  (vlan10)| |(vlan10)  laptop1 laptop2 | | |
| Description | Lan port works with port based mode, check ingress packet vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR lan port works with port based mode | | |
| Test procedure | 1. configure mgt0.1 vlan 10 2. enable mgt0.1 dhcp, dns server 3. configure mgt0.1 ip address as 10.10.10.1, ip-pool as 10.10.10.10-10.10.10.99 4. configure mgt0.2 vlan 20 5. enable mgt0.2 dhcp, dns server 6. configure mgt0.2 ip address as 20.20.20.1, ip-poop as 20.20.20.20-20.20.20.99 7. ***Lan port eth1/eth4 in same vlan with vlan10*** 8. laptop1 connect to lan port eth1 9. laptop2 connect to lan port eth4 10. Check laptop1, laptop2’s ip ***Result1*** 11. Check mac-table ***Result2***   ***#show lan mac-table***  ***#show interface eth1 mac-learning all***  ***#show interface eth4 mac-learning all***   1. Laptop1 send a packet without vlan tag to laptop2. (laptop1 ping laptop2)   -check if laptop2 receive the packet. ***Result3***  -check if the received packet has vlan tag.  ***Result4***   1. Laptop1 send a packet with vlan tag to laptop2, the packet vlan is same with the port default vlan (vlan10)   -check if laptop2 receive the packet. ***Result5***  -check if the received packet has vlan tag. ***Result6***   1. Laptop1 send a packet with vlan tag to laptop2, the packet vlan is different with the port default vlan   Laptop1 send a packet with vlan tag 20 ***Result7*** | | |
| Expect result | ***Result1:*** laptop1, laptop2 get ip successfully from mgt0.1 dhcp server    ***Result2:*** mac-table has entries for laptop1, laptop2  ***Result3:*** laptop1 ping laptop2 successfully, laptop2 can receive the packet  ***local forwarding by switch chip***  ***Result4:*** the received packet don’t have vlan tag [vlan tag is stripped when egress]  ***Result5:*** laptop2 receive the packet successfully  ***local forwarding by switch chip***  ***Result6***: the received packet don’t have vlan tag [vlan tag is stripped when egress]  ***Result7:***  laptop2 can’t receive the packet  Hw driver will drop the packet  [switch chip can learn the mac address, the packet drop by hw, so FE can’t learn, check “show interface eth1 mac-learning all”] | | |
| Test result | BR100: pass **[Result7: bug15069 due to hw limitation]**  BR200: bug16470  Result7结果不对 in access mode we do not have vlan table in  switch. the chip can't drop this packet. | | |
| Comment |  | | |

##### Packets forwarding with/without vlan tag (lan to wan/wifi)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_PortBased\_Vlan\_3 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)----(wifi)S1 (vlan1)  (vlan10) | \ (wifi)S2 (vlan10)  Laptop1 | | |
| Description | Lan port works with port based mode, designate vlan for lan ports, check lan to wan/wifi/AP forwarding  ***[L3 route forwarding]*** | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  S1,S2 connect to AP via wifi  S1 with vlan 1  Laptop1 and S2 with same vlan 10  BR lan port works with port based mode | | |
| Test procedure | 1. Configure mgt0.1 vlan 10, enable dhcp, dns server, configure ip address, ip-pool   Configure mgt0.1,  mgt0.1 ip as 10.10.10.1  mgt0.1 vlan as 10  ip-pool as 10.10.10.10-10.10.10.99  Configure lan ports vlan as 10   1. Check lan port’s mode, vlan id info.  ***Result1*** 2. Laptop1 connect to eth1, check mac-table. ***Result2*** 3. Check laptop1’s ip address. ***Result3***   ***Laptop1 and S2 with same vlan 10***   1. Laptop1 send a packet to S2. Laptop1 ping S2. ***Result4***   ***Laptop1, PC1, S1 are in different subnet***   1. Laptop1 send a packet to PC1/S1/mgt0.1 without vlan tag. (ping) ***Result5*** 2. Laptop1 send a packet to PC1/S1/mgt0.1 with a vlan tag same with lan port’s vlan id. ***Result6*** 3. Laptop1 send a packet to PC1/S1/mgt0.1 with a vlan tag not same with lan port’s vlan id. ***Result7*** | | |
| Expect result | ***Result1:***  lan port’s mode is port based, vlan id is 10  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***Result2:*** mac-table has an entry for laptop1  ***# show lan mac-table***  ***Result3:*** laptop1’s ip address is assigned by mgt0.1 dhcp server, ip address is 10.10.10.10  ***Result4:*** laptop1 ping S2 successfully  ***Result5:*** ---laptop1 ping pc1 successfully, pc1 can receive the packet without vlan tag.  ---Laptop1 ping S1 successfully, s1 can receive the packet without vlan tag  ---Laptop1 ping mgt0.1 successfully, mgt0.1 can receive the packet without vlan tag  ***Result6***: PC1 can receive the packet without vlan tag successfully  Mgt0.1 can receive the packet without vlan tag successfully  ***Result7:*** BR will drop the packet by switch chip, since the packet vlan tag is not same with port’s vlan | | |
| Test result | BR100: pass  BR200: access still can receive [Result7: bug16470 ] | | |
| Comment |  | | |

#### Trunk mode vlan

##### Trunk ports default vlan (mgt0 native-vlan)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_1 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)－laptop1 | | |
| Description | Lan ports work with trunk mode, check lan ports native-vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1  Lan ports work with trunk mode | | |
| Test procedure | 1. Enable mgt0 dhcp/dns server 2. Lan port works with trunk mode, check lan ports mode, vlan ID.  ***Result1*** 3. Check allowed vlan list, vlan-id-check flag. ***Result2*** 4. Laptop1 connect to eth1, check HW ARL table (mac-table). ***Result3*** 5. Check laptop ip . ***Result4*** 6. Configure mgt0 native-vlan as 10 7. Check lan ports native-vlan ***Result5*** 8. Configure mgt0.1 vlan 10, enable dhcp, dns server, configure ip address, ip-pool   ***mgt0.1 ip as 10.10.10.1***  ***mgt0.1 vlan as 10***  ***ip-pool as 10.10.10.10-10.10.10.99***   1. Laptop1 connect to eth1, check laptop1 ip address  ***Result6*** | | |
| Expect result | ***Result1:*** the default lan ports mode is trunk, and default vlan is 1.  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***Result2:*** default has no allowed vlan list, vlan-id-check flag is No  ***Result3:*** mac-table has an entry for laptop1, indicates port id, laptop mac address  ***#show lan mac-table***  ***Result4:***  laptop get an ip address successfully, which is assigned by mgt0’s dhcp server  Ip address is 192.168.85.10  ***Result5:*** lan ports native vlan is 10 mapped from mgt0’s native vlan  ***Result6:***  laptop1 ip address is assigned by mgt0.1 dhcp server  Laptop1 ip address is 10.10.10.10 | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Trunk ports default vlan (per interface native-vlan)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_2 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)－laptop1 | | |
| Description | Lan ports work with trunk mode, check per interface native-vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1  Lan ports work with trunk mode | | |
| Test procedure | 1. Enable mgt0 dhcp/dns server 2. Lan port works with trunk mode, check lan ports mode, vlan ID.  ***Result1*** 3. Check allowed vlan list, vlan-id-check flag. ***Result2*** 4. Laptop1 connect to eth1, check HW ARL table (mac-table). ***Result3*** 5. Check laptop ip . ***Result4*** 6. Configure mgt0 native-vlan as 10 7. Check lan ports native-vlan ***Result5*** 8. Configure mgt0.1 vlan 10, enable dhcp, dns server, configure ip address, ip-pool 9. Laptop1 connect to eth1, check laptop1 ip address  ***Result6*** 10. Configure eth1 native-vlan 20   ***Interface eth1 native-vlan 20***   1. Configure mgt0.2 vlan 20, enable dhcp, dns server, configure ip address, ip-pool 2. Check lan port eth1 native-vlan ***Result7*** 3. Check laptop1 ip address ***Result8*** | | |
| Expect result | ***Result1:*** the default lan ports mode is trunk, and default vlan is 1.  ***#show lan port-config***  ***#show lan vlan-table***  ***#show interface***  ***Result2:*** default has no allowed vlan list, vlan-id-check flag is No  ***Result3:*** mac-table has an entry for laptop1, indicates port id, laptop mac address  ***#show lan mac-table***  ***Result4:***  laptop get an ip address successfully, which is assigned by mgt0’s dhcp server  Ip address is 192.168.85.10  ***Result5:*** lan ports native vlan is 10 mapped from mgt0’s native vlan  ***Result6:***  laptop1 ip address is assigned by mgt0.1 dhcp server  Laptop1 ip address is 10.10.10.10  ***Result7:***  lan port eth1 native-vlan is 20, lan ports eth2/eth3/eth4 native-vlan is 10  ***Result8:*** laptop1 ip address is assigned by mgt0.2 dhcp server  Laptop1 ip address is 20.20.20.20  Priority of native-vlan mapping: ethx native-vlan > mgt0 native-vlan | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Trunk ports allowed vlan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_3 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch1  |  (eth0)BR(eth1-eth4)－laptop1 (native-vlan10)  |trunk(native-vlan 10, allow-vlan 20)  Pc1 | | |
| Description | Lan ports work with trunk mode, check allowed vlan | | |
| Pre-condition | Pc1 connect to switch2  Laptop1 connect to lan port eth4 with access mode  Switch2 connect to lan port eth1 with trunk mode  Pc1 connect to switch2  Lan ports work with trunk mode | | |
| Test procedure | 1. Configure mgt0.1 vlan 10, enable dhcp/dns server, ip-pool is 10.10.10.10-10.10.10.99   Configure mgt0.2 vlan20, enable dhcp/dns server, ip-pool is 20.20.20.20-20.20.20.99  Configure mgt0.2 vlan 30, enable dhcp/dns server, ip-pool is 30.30.30.30-30.30.30.99   1. Configure native-vlan for lan ports as vlan 10   CLI:  Interface mgt0 native-vlan 10   1. Connect pc1 to lan port eth1 2. check ip address of pc1  ***Result1*** 3. Configure lan port eth1 native-vlan 10, allowed-vlan 20   **Interface eth1 allow-vlan 10**  **Interface eth1 allow-vlan 20**   1. Configure lan port eth4 native-vlan as 10 with access mode 2. check vlan-table. ***Result2*** 3. PC1 send packet with vlan tag 10 to laptop1 ***Result3*** 4. Change lan port eth4 native-vlan as vlan 20.   **Interface eth4 mode bridge-access user-profile-attribute 20**   1. Pc1 send packet with vlan tag 20 to laptop1 ***Result4*** 2. Lan port eth4 native-vlan as vlan 30.   **Interface eth4 mode bridge-access user-profile-attribute 30**   1. Pc1 send packet with vlan tag 30 to laptop1 ***Result5*** 2. add an allowed-vlan 30 for lan port eth1   interface eth1 allow-vlan 30   1. pc1 send packet with vlan tag 30 to laptop2 ***Result6*** | | |
| Expect result | ***Result1:*** pc1/laptop1 obtain ip address from mgt0.1 dhcp server, ip address is 10.10.10.10/10.10.10.11  ***Result2:*** vlan-table has entries for eth1 allowed-vlan, and the vlan-id-check for eth1 is enabled  Port-config table has entries for eth1/eth4 native-vlan  ***#show lan vlan-table***  ***#show lan port-config***  ***AH-5a3e40#show lan vlan-table***  ***eth1 eth2 eth3 eth4 VLAN***  ***--------------------------------------------***  ***eth1 eth2 eth3 eth4 10***  ***eth1 20***  ***AH-5a3e40#show lan port-config***  ***Port Link Open Mode Default VLAN VLAN Check***  ***-----------------------------------------------------------------------***  ***eth1 U Y trunk 10 Yes***  ***eth2 U Y trunk 10 No***  ***eth3 U Y trunk 10 No***  ***eth4 U Y port-based 10 -***  ***Result3:*** laptop1 can receive the packet  ***Result4:***  laptop1 can receive the packet  ***Result5:*** laptop1 can’t receive the packet; the packet is dropped by hw.  ***Result6:*** laptop1 can receive the packet | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Packet forwarding with/without vlan tag

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_4 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)－laptop1  |  Laptop2 | | |
| Description | Lan ports work with trunk mode, check ingress packet forwarding with/without vlan tag | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  Lan ports work with trunk mode | | |
| Test procedure | 1. Lan port eth1/eth4 in same vlan 2. Configure native-vlan, allowed-vlan list for ethx 3. Laptop1 send a packet to laptop2 without vlan tag. (icmp) ***Result1*** 4. Laptop1 send a packet to laptop2 with a vlan tag match native-vlan. (icmp) ***Result2*** 5. Laptop1 send a packet to laptop2 with a vlan tag match allowed-vlan list. (icmp) ***Result3*** 6. *Laptop1 send a packet tolaptop2 with a vlan tag not match allowed-vlan list. (icmp)*   ***Result4*** | | |
| Expect result | ***Result1:*** laptop2 will receive the packet, but the receive request pkt has native-vlan tag [it’s hw limitation]  .( sw driver will add the native-vlan tag for the packet, retrieve the native-vlan id, write to skb->vlan\_tci field)  ***Result2:*** laptop2 will receive the packet. [egress pkts also have native-vlan tag]  ( retrieve the packets vlan id, write to skb->vlan\_tci field) strip tag  ***Result3:*** laptop2 can’t receive the packet  ( since the lan port eth4 vlan is not the same as the packet vlan tag)  ***Result4:***  laptop2 can’t receive the packet. (since the packet is dropt by switch chip directly) | | |
| Test result | BR100: pass [For Result1/Result2, egress pkt has (egress port) native-vlan tag]  BR200: pass | | |
| Comment |  | | |

##### Packet forwarding with vlan-id-check

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_5 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)－laptop1  |  Laptop2 | | |
| Description | Lan ports work with trunk mode, check ingress packet forwarding with vlan-id-check enabled | | |
| Pre-condition | Pc1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  Lan ports work with trunk mode | | |
| Test procedure | 1. Configure native-vlan, allowed-vlan list for eth1   Native-vlan as 10: interface mgt0 native-vlan 10 / interface th1 native-vlan 10   1. ***Disable vlan-id-check feature for eth1***   CLI:  No interface eth1 allowed-vlan   1. Lan port eth4 is access mode with native-vlan 30 2. Check vlan-table/port-config table ***Result1*** 3. Laptop1 send a packet to laptop2 with vlan tag 30, not match native-vlan and allowed-vlan list. (icmp) ***Result2*** 4. ***Enable vlan-id-check feature for eth1***  ***Result3***   CLI:  Interface eth1 allowed-vlan auto   1. Laptop1 send a packet to laptop2 with vlan tag 30, but not match native-vlan and allowed-vlan list. (icmp) ***Result3*** | | |
| Expect result | ***Result1:***  A***H-5a3e40#show lan port-config***  ***Port Link Open Mode Default VLAN VLAN Check***  ***-----------------------------------------------------------------------***  ***eth1 U Y trunk 10 No***  ***eth2 U Y trunk 10 Yes***  ***eth3 U Y trunk 10 Yes***  ***eth4 D Y port-based 30 -***  ***AH-5a3e40#show lan vlan-table***  ***eth1 eth2 eth3 eth4 VLAN***  ***--------------------------------------------***  ***eth1 20***  ***eth4 30***  ***eth1 eth2 eth3 10***  ***Result2:*** laptop2 will receive the packet.  ***Result3:***  ***AH-5a3e40#show lan port-config***  ***Port Link Open Mode Default VLAN VLAN Check***  ***-----------------------------------------------------------------------***  ***eth1 U Y trunk 10 Yes***  ***eth2 U Y trunk 10 Yes***  ***eth3 D Y trunk 10 Yes***  ***eth4 U Y port-based 30 -***    ***Result4:*** laptop2 can’t receive the packet.  (the vlan-id-check enabled, if not match vlan list, HW drop the packet.) | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment | ***BR100: Allow vlan list support 16 vlan(include native-vlan).***  ***Interface ethx allow-vlan all***  ***BR200: allow vlan list support 4094 vlan*** | | |

##### Egress packet with trunk mode

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Trunk\_Vlan\_6 | | |
| Priority | Accept | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)－laptop1  |  (AP/BR) | | |
| Description | Lan ports work with trunk mode, check the egress packet vlan tag | | |
| Pre-condition | Pc1 connect to switch  Laptop1 connect to lan port eth4  AP connect to lan port eth1  Lan ports work with trunk mode | | |
| Test procedure | 1. Configure eth1/eth4 as trunk mode 2. Configure its allowed-vlan list and native-vlan:   native-vlan is 10, allowed-vlan as 20  CLI:  interface mgt0 native-vlan 10  interface eth1 allowed-vlan 10  interface eth1 allowed-vlan 20  interface eth4 allowed-vlan 20   1. Configure mgt0.1 for vlan 10, mgt0.2 for vlan 20 2. Check AP and laptop1 ip address. [network card of laptop1 support capture vlan-tag] ***Result1*** 3. Laptop1 send a packet to AP, the egress packet vlan id is match native-vlan, check packet vlan tag received by AP. (laptop1 send icmp packet to AP using tool) ***Result2*** 4. Laptop1 send a packet to AP, the egress packet vlan id is match allowed-vlan list. Check packet vlan tag received by AP. (laptop1 send icmp packet to AP using tool) ***Result3*** | | |
| Expect result | ***Result1:***  AP and laptop1 can get ip address from mgt0.1 dhcp server successfully.  **-----BR100:**  **Failed. Bug15388: AP/Laptop1 can’t get ip address, since the egress packet will be with native-vlan tag**  **-----BR200: pass**  ***Result2:***  capture packet on AP, the packet doesn’t have vlan tag.  (the vlan tag is stripped when transmitted through eth4)  **Failed. Bug15388: AP/Laptop1 can’t get ip address, since the egress packet will be with native-vlan tag**  ***Result3:*** capture packet on AP, the packet has vlan tag. | | |
| Test result | BR100:pass **[Bug15388 not fixed, because of hw limitation.**  Further, Atheros has already told us they do not support the trunk mode (untagged native vlan) on AR9331. **]**  BR200: pass | | |
| Comment | BR200 still has bug16488, fixed already | | |

### Lan ports datapath

##### Known unicast datapath

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_ 1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)-------(wifi)S1  | | \(wifi)S2  Laptop1 Laptop2 | | |
| Description | Check **known unicast** packet forwarding path | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth2  Station1 connect to wifi  Station2 connect to wifi  Pc1, laptop1, s1 with different subnet  Laptop1,laptop2, s2 with same subnet  BR lan port works with port based / trunk mode | | |
| Test procedure | 1. Laptop1, laptop2 in same vlan 2. Laptop1, laptop2 can retrieve ip address. ***Result1*** 3. Station1 connect to BR via wifi, pass authentication.  ***Result2***   **Local forwarding by hw**   1. Laptop1 send unicast packet to laptop2, like ping. ***Result3*** 2. Laptop2 send unicast packet to laptop1, like ping. ***Result4***   **L3 route forwarding**  **lan/wifi with different subnet**   1. Laptop1/laptop2 send unicast packet to S1, like ping. ***Result5*** 2. S1 send unicast packet to laptop1/laptop2, like ping. ***Result6***   **lan/wifi with same subnet**   1. Laptop1/laptop2 send unicast packet to s2, like ping. ***Result7*** 2. S2 send unicast packet to laptop1/laptop2, like ping. ***Result8***   **L3 route forwarding**   1. Laptop1/laptop2 send unicast packet to pc1, like ping. ***Result9*** 2. Pc1 send unicast packet to laptop1/laptop2, like ping. ***Result10*** | | |
| Expect result | ***Result1:*** retrieve ip from dhcp server successfully  ***Result2:*** station1 associated to BR successfully  ***Result3:***  ping successfully, route-lookup, send to laptop2 directly.  ***Local forwarding packets among lan ports***  ***Result4:*** ping successfully, route-lookup, send to laptop1 directly.  ***Local forwarding packets among lan ports***  ***Result5:*** laptop ping station1 successfully  ***Result6:*** station1 ping laptop successfully  ***Result7:*** laptop ping S2 successfully  ***Result8:*** S2 ping laptop successfully  ***Result9:*** laptop ping pc1 successfully  ***Result10:*** pc1 ping laptop successfully | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Unknown unicast datapath

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_ 2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)--------(wifi)Station1  | |  Laptop1 Laptop2 | | |
| Description | Check **unknown unicast** packet forwarding path | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  Laptop1 and laptop2 with same vlan  Station1 connect to wifi  BR lan port works with port based/trunk mode | | |
| Test procedure | 1. On pc1, laptop1/laptop2 create a static arp entry, such as 10.155.30.222 -> 00-11-22-33-44-55 2. Unknown unicast from eth0 interface   Pc1 ping the created static ip address. ***Result1***   1. Unknown unicast from lan ports eth1 interface   Laptop1 ping the created static ip address. ***Result2*** | | |
| Expect result | ***Result1:***  BR will drop the packet  ***Result2:***  BR will drop the packet | | |
| Test result |  | | |
| Comment |  | | |

##### Broadcast datapath with port based mode (packet forwarding)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_3 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)--------(wifi)Station1  | |  Laptop1 Laptop2 | | |
| Description | Check broadcast packet forwarding path with port based mode | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth2  Station1 connect to wifi  BR lan port works with port based mode  Lan ports are in same vlan | | |
| Test procedure | 1. Pc1 send broadcast packet, like arp request. ***Result1*** 2. Station1 send broadcast packet, like arp request. ***Result2*** 3. Laptop1/laptop2 send broadcast packet, like arp request. ***Result3*** | | |
| Expect result | ***Result1:***  the broadcast packet from pc1 will be dropped  Action: PC1 ping an unexist ip address  Debug info:  2012-01-09 05:31:52 debug kernel: (i) eth0 d067:e531:4df9->ffff:ffff:ffff arp 10.155.30.16(d067:e531:4df9)->10.155.30.100(0000:0000:0000) 46 bytes  2012-01-09 05:31:52 debug kernel: [fe]: bypass fe ingress procesing and deliver packet to stack on eth0  2012-01-09 05:31:52 debug kernel: [fe]: routing done, 77.249.10.155 -> 30.16.0.0 ttl(208) proto(103) mark(0) 46 bytes, found route in table:Main(Split)  ***Result2:***  the broadcast packet from wifix will relay to lan ports(up), wifix, mgt0  Debug info:  Don’t relay to eth0  2012-01-09 05:37:35 debug kernel: (i) wifi0.2 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 28 bytes  2012-01-09 05:37:35 debug kernel: [fe]: MAC session (id [4]) found  2012-01-09 05:37:35 debug kernel: [fe]: fflow 001a:921c:6e44->ffff:ffff:ffff flag 0x40600, rflow ffff:ffff:ffff->001a:921c:6e44 flag 0xa1810  2012-01-09 05:37:35 debug kernel: [fe]: fflow acl 0x0/0x0, rflow acl 0x0/0x0  2012-01-09 05:37:35 debug kernel: [fe]: ARP REQ return code 0, no action  2012-01-09 05:37:35 debug kernel: [fe]: QoS: ingress pkt fwd(wifi0.2) profile=111 qos=2  2012-01-09 05:37:35 debug kernel: [fe]: flood the packet out wifi0.3, skb size 288, len 28, data\_len 0, head room 124, tail room 136  2012-01-09 05:37:35 debug kernel: [fe]: wifi0.3 Tx 001a:921c:6e44 -> ffff:ffff:ffff type 0x0806 42 bytes  2012-01-09 05:37:35 debug kernel: (o) wifi0.3 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:37:35 debug kernel: [fe]: flood the packet out wifi0.2, skb size 288, len 28, data\_len 0, head room 124, tail room 136  2012-01-09 05:37:35 debug kernel: [fe]: wifi0.2 Tx 001a:921c:6e44 -> ffff:ffff:ffff type 0x0806 42 bytes  2012-01-09 05:37:35 debug kernel: (o) wifi0.2 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:37:35 debug kernel: [fe]: flood the packet out wifi0.1, skb size 288, len 28, data\_len 0, head room 124, tail room 136  2012-01-09 05:37:35 debug kernel: [fe]: set nhop ffff:ffff:ffff ttl 15  2012-01-09 05:37:35 debug kernel: [fe]: wifi0.1 Tx 001a:921c:6e44 -> ffff:ffff:ffff type 0x0806 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: wifi0.1 stuff ah-hdr nhop ffff:ffff:ffff ttl 15  2012-01-09 05:37:35 debug kernel: (o) wifi0.1 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:37:35 debug kernel: [fe]: mgt0.3 is sub-interface, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: mgt0.2 is sub-interface, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: mgt0.1 is sub-interface, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: interface down, don't flood on eth4sniffer  2012-01-09 05:37:35 debug kernel: [fe]: interface down, don't flood on eth4  2012-01-09 05:37:35 debug kernel: [fe]: interface down, don't flood on eth3sniffer  2012-01-09 05:37:35 debug kernel: [fe]: interface down, don't flood on eth3  2012-01-09 05:37:35 debug kernel: [fe]: interface down, don't flood on eth2sniffer  2012-01-09 05:37:35 debug kernel: [fe]: flood the packet out eth2, skb size 288, len 28, data\_len 0, head room 124, tail room 136  2012-01-09 05:37:35 debug kernel: [fe]: eth2 Tx 001a:921c:6e44 -> ffff:ffff:ffff type 0x0806 42 bytes  2012-01-09 05:37:35 debug kernel: (o) eth2 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: QoS: pkt forwarded  2012-01-09 05:37:35 debug kernel: [fe]: eth1sniffer is sniffer interface, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: flood the packet out eth1, skb size 288, len 28, data\_len 0, head room 124, tail room 136  2012-01-09 05:37:35 debug kernel: [fe]: eth1 Tx 001a:921c:6e44 -> ffff:ffff:ffff type 0x0806 42 bytes  2012-01-09 05:37:35 debug kernel: (o) eth1 001a:921c:6e44->ffff:ffff:ffff arp 10.10.10.15(001a:921c:6e44)->10.10.10.99(0000:0000:0000) 42 bytes  2012-01-09 05:37:35 debug kernel: [fe]: QoS: pkt forwarded  2012-01-09 05:37:35 debug kernel: [fe]: route? yes, mesh point 0019:775a:3e40 (mac 001a:921c:6e44) not owned on dev reth1, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: eth0sniffer is sniffer interface, don't flood  2012-01-09 05:37:35 debug kernel: [fe]: swap incoming dev wifi0.2 -> mgt0.1  2012-01-09 05:37:35 debug kernel: [fe]: dst-ip 10.10.10.99 doesn't match ip device (mgt0.1) IP, drop pkt  2012-01-09 05:37:35 debug kernel: [fe]: deliver pak to self on eth1sniffer with fw mark 0  2012-01-09 05:37:35 debug kernel: (u) eth1sniffer 001a:921c:6e44->ffff:ffff:ffff(0x8100) 46 bytes  2012-01-09 05:37:35 debug kernel: [fe]: Tx:wifi0.3:0> 001a:921c:6e44->ffff:ffff:ffff profile idx=1 pkt\_len=42 q\_len=1 QoS buf=1904  2012-01-09 05:37:35 debug kernel: [fe]: Tx:wifi0.2:0> 001a:921c:6e44->ffff:ffff:ffff profile idx=1 pkt\_len=42 q\_len=0 QoS buf=952  2012-01-09 05:37:35 debug kernel: [fe]: Tx:wifi0.1:0> 001a:921c:6e44->ffff:ffff:ffff profile idx=1 pkt\_len=42 q\_len=0 QoS buf=0  2012-01-09 05:37:35 debug kernel: [fe]: deliver pak to self on wifi0sniffer with fw mark 0  2012-01-09 05:37:35 debug kernel: (u) wifi0sniffer 0100:f4ca:97b2->0000:1b00:0f8c LLC 95 bytes  2012-01-09 05:37:35 debug kernel: [fe]: mark pkt as from self  2012-01-09 05:37:35 debug kernel: [fe]: inject pkt back into flow from mgt0 xmit  ***Result3:***  **BR100 result:**  the broadcast packet from laptopx will relay to wifix, lan ports(up), mgt0  Don’t flood to eth0  2012-01-09 05:43:06 debug kernel: (i) eth3 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 46 bytes  2012-01-09 05:43:06 debug kernel: [fe]: MAC session (id [1]) found  2012-01-09 05:43:06 debug kernel: [fe]: fflow d067:e531:4df9->ffff:ffff:ffff flag 0x40600, rflow ffff:ffff:ffff->d067:e531:4df9 flag 0x81810  2012-01-09 05:43:06 debug kernel: [fe]: fflow acl 0x0/0x0, rflow acl 0x0/0x0  2012-01-09 05:43:06 debug kernel: [fe]: ARP REQ return code 0, no action  2012-01-09 05:43:06 debug kernel: [fe]: QoS: ingress pkt fwd(eth3) profile=0 qos=2  2012-01-09 05:43:06 debug kernel: [fe]: flood the packet out wifi0.3, skb size 320, len 46, data\_len 0, head room 116, tail room 158  2012-01-09 05:43:06 debug kernel: [fe]: wifi0.3 Tx d067:e531:4df9 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-01-09 05:43:06 debug kernel: (o) wifi0.3 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:43:06 debug kernel: [fe]: flood the packet out wifi0.2, skb size 320, len 46, data\_len 0, head room 116, tail room 158  2012-01-09 05:43:06 debug kernel: [fe]: wifi0.2 Tx d067:e531:4df9 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-01-09 05:43:06 debug kernel: (o) wifi0.2 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:43:06 debug kernel: [fe]: flood the packet out wifi0.1, skb size 320, len 46, data\_len 0, head room 116, tail room 158  2012-01-09 05:43:06 debug kernel: [fe]: set nhop ffff:ffff:ffff ttl 15  2012-01-09 05:43:06 debug kernel: [fe]: wifi0.1 Tx d067:e531:4df9 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: wifi0.1 stuff ah-hdr nhop ffff:ffff:ffff ttl 15  2012-01-09 05:43:06 debug kernel: (o) wifi0.1 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: QoS: pkt queued  2012-01-09 05:43:06 debug kernel: [fe]: mgt0.3 is sub-interface, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: mgt0.2 is sub-interface, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: mgt0.1 is sub-interface, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: interface down, don't flood on eth4sniffer  2012-01-09 05:43:06 debug kernel: [fe]: interface down, don't flood on eth4  2012-01-09 05:43:06 debug kernel: [fe]: interface down, don't flood on eth3sniffer  2012-01-09 05:43:06 debug kernel: [fe]: eth3 no broadcast out from incoming port  2012-01-09 05:43:06 debug kernel: [fe]: interface down, don't flood on eth2sniffer  2012-01-09 05:43:06 debug kernel: [fe]: flood the packet out eth2, skb size 320, len 46, data\_len 0, head room 116, tail room 158  2012-01-09 05:43:06 debug kernel: [fe]: eth2 Tx d067:e531:4df9 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-01-09 05:43:06 debug kernel: (o) eth2 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: QoS: pkt forwarded  2012-01-09 05:43:06 debug kernel: [fe]: eth1sniffer is sniffer interface, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: flood the packet out eth1, skb size 320, len 46, data\_len 0, head room 116, tail room 158  2012-01-09 05:43:06 debug kernel: [fe]: eth1 Tx d067:e531:4df9 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-01-09 05:43:06 debug kernel: (o) eth1 d067:e531:4df9->ffff:ffff:ffff arp 10.10.10.10(d067:e531:4df9)->10.10.10.99(0000:0000:0000) 60 bytes  2012-01-09 05:43:06 debug kernel: [fe]: QoS: pkt forwarded  2012-01-09 05:43:06 debug kernel: [fe]: route? yes, mesh point 0019:775a:3e40 (mac d067:e531:4df9) not owned on dev reth1, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: eth0sniffer is sniffer interface, don't flood  2012-01-09 05:43:06 debug kernel: [fe]: swap incoming dev eth3 -> mgt0.1  2012-01-09 05:43:06 debug kernel: [fe]: dst-ip 10.10.10.99 doesn't match ip device (mgt0.1) IP, drop pkt  2012-01-09 05:43:06 debug kernel: [fe]: Tx:wifi0.3:0> d067:e531:4df9->ffff:ffff:ffff profile idx=0 pkt\_len=60 q\_len=2 QoS buf=1968  2012-01-09 05:43:06 debug kernel: [fe]: Tx:wifi0.2:0> d067:e531:4df9->ffff:ffff:ffff profile idx=0 pkt\_len=60 q\_len=1 QoS buf=984  2012-01-09 05:43:06 debug kernel: [fe]: Tx:wifi0.1:0> d067:e531:4df9->ffff:ffff:ffff profile idx=0 pkt\_len=60 q\_len=0 QoS buf=0  **BR200 result:**  the broadcast packet from laptopx will relay to wifix, mgt0, and lan ports(local swich by switch chip), don’t flood to eth0  #2012-02-03 02:39:37 debug kernel: L\*: (i) eth2 0015:c5c3:5f21->ffff:ffff:ffff arp 192.168.85.10(0015:c5c3:5f21)->192.168.85.88(0000:0000:0000) 46 bytes  2012-02-03 02:39:37 debug kernel: [fe]: MAC session (id [3]) found  2012-02-03 02:39:37 debug kernel: [fe]: fflow 0015:c5c3:5f21->ffff:ffff:ffff flag 0x40600, rflow ffff:ffff:ffff->0015:c5c3:5f21 flag 0x81810  2012-02-03 02:39:37 debug kernel: [fe]: fflow acl 0x0/0x0, rflow acl 0x0/0x0  2012-02-03 02:39:37 debug kernel: [fe]: ARP REQ return code 0, no action  2012-02-03 02:39:37 debug kernel: [fe]: QoS: ingress pkt fwd(eth2) profile=0 qos=2  2012-02-03 02:39:37 debug kernel: [fe]: flood the packet out wifi0.2, skb size 288, len 46, data\_len 0, head room 86, tail room 156  2012-02-03 02:39:37 debug kernel: [fe]: wifi0.2 Tx 0015:c5c3:5f21 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-02-03 02:39:37 debug kernel: L\*: (o) wifi0.2 0015:c5c3:5f21->ffff:ffff:ffff arp 192.168.85.10(0015:c5c3:5f21)->192.168.85.88(0000:0000:0000) 60 bytes  2012-02-03 02:39:37 debug kernel: [fe]: QoS: pkt queued  2012-02-03 02:39:37 debug kernel: [fe]: flood the packet out wifi0.1, skb size 288, len 46, data\_len 0, head room 86, tail room 156  2012-02-03 02:39:37 debug kernel: [fe]: set nhop ffff:ffff:ffff ttl 15  2012-02-03 02:39:37 debug kernel: [fe]: wifi0.1 Tx 0015:c5c3:5f21 -> ffff:ffff:ffff type 0x0806 60 bytes  2012-02-03 02:39:37 debug kernel: [fe]: wifi0.1 stuff ah-hdr nhop ffff:ffff:ffff ttl 15  2012-02-03 02:39:37 debug kernel: L\*: (o) wifi0.1 0015:c5c3:5f21->ffff:ffff:ffff arp 192.168.85.10(0015:c5c3:5f21)->192.168.85.88(0000:0000:0000) 60 bytes  2012-02-03 02:39:37 debug kernel: [fe]: QoS: pkt queued  2012-02-03 02:39:37 debug kernel: [fe]: BR200(local switch) skip broadcast eth2 -> eth4  2012-02-03 02:39:37 debug kernel: [fe]: BR200(local switch) skip broadcast eth2 -> eth3  2012-02-03 02:39:37 debug kernel: [fe]: eth2 no broadcast out from incoming port  2012-02-03 02:39:37 debug kernel: [fe]: BR200(local switch) skip broadcast eth2 -> eth1  2012-02-03 02:39:37 debug kernel: [fe]: BR200(local switch) skip broadcast eth2 -> lan  2012-02-03 02:39:37 debug kernel: [fe]: BR200(local switch) skip broadcast eth2 -> eth0sniffer  2012-02-03 02:39:37 debug kernel: [fe]: swap incoming dev eth2 -> mgt0  2012-02-03 02:39:37 debug kernel: [fe]: dst-ip 192.168.85.88 doesn't match ip device (mgt0) IP, drop pkt  2012-02-03 02:39:37 debug kernel: [fe]: Tx:wifi0.2:0> 0015:c5c3:5f21->ffff:ffff:ffff profile idx=0 pkt\_len=60 q\_len=1 QoS buf=960  2012-02-03 02:39:37 debug kernel: [fe]: Tx:wifi0.1:0> 0015:c5c3:5f21->ffff:ffff:ffff profile idx=0 pkt\_len=60 q\_len=0 QoS buf=0 | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### Broadcast datapath with trunk mode----important

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_4 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth3)---wifix(sta1)  |10 |20 |20  Laptop1 Laptop2 Laptop3 | | |
| Description | Check broadcast packet forwarding path with trunk mode | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth2  Laptop3 connect to lan port eth3  BR lan port eth1/eth2 works with trunk mode  BR lan port eth3 works with port-based mode  Lan port eth1, eth2 ,eth3 with different vlan | | |
| Test procedure | 1. Lan port eth1 trunk mode with vlan 10, disable vlan-id-check 2. Lan port eth2 trunk mode with vlan 20, allow-vlan 20,40 3. Lan port eth3 acess mode with vlan 30 4. Create mgt0.5 dhcp server, with vlan 50   *interface mgt0.5 vlan 50*  *interface mgt0.5 ip 50.50.50.1/24*  *interface mgt0.5 dhcp-server options default-gateway 50.50.50.1*  *interface mgt0.5 dhcp-server ip-pool 50.50.50.50 50.50.50.99*  *interface mgt0.5 dhcp-server enable*  *interface mgt0.5 dns-server enable*   1. Laptop1 send broadcast packet with vlan tag 10.   Capture packet on laptop2, laptop3. ***Result1***   1. Laptop1 send broadcast packet with vlan tag 20.   Capture packet on laptop2, laptop3. ***Result2***   1. Sta1 associated with vlan 50 2. Sta1 Ping an unexist ip for vlan 50 on BR   #Ping 50.50.50.80  Capture packet on eth1/eth2/eth3 ***Result3***   1. Add allow-vlan 50 for lan port eth2 2. Sta1 Ping an unexist ip for 50 on BR 3. Capture packet on laptop1/laptop2/laptop3 ***Result4*** | | |
| Expect result | ***Result1:*** can’t capture packet on laptop2/laptop3*.*  Action: use tool to send broadcast packet with vlan 10  ***Result2:***  laptop2 can capture the packet, laptop3 can’t capture the packet  Action: use tool to send broadcast packet with vlan 20  ***Result3:***  laptop1 can receive the packet with vlan tag 50 arp pkt, laptop2/laptop3 can’t receive the packet [support vlan separation from cpu]  ***Result4:*** laptop1/laptop2 can receive the packet with vlan tag 50 pkt, laptop3 can’t receive. | | |
| Test result | BR100: pass Result3: bug16146  BR200: pass | | |
| Comment |  | | |

##### Dhcp packet datapath(laptop connect to ethx)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_5 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth1-eth4)--------(wifi)Station1  | |  Laptop1 Laptop2 | | |
| Description | Check dhcp packet process for laptop1/laptop2 with lan ports port based mode | | |
| Pre-condition | PC1 connect to switch  Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  Station1 connect to wifi  BR lan port works with port based/trunk mode | | |
| Test procedure | 1. Configure mgt0.1 vlan 20 2. Configure user-profile vlan 20 attribute 20   User-profile test20 vlan 20 attribute 20   1. Configure lan port eth1/eth4 user-profile-attribute 20   Interface eth1 mode bridge-access user-profile-attribute 20  Interface eth4 mode bridge-access user-profile-attribute 20   1. Laptop1 connect to eth1, retrieve ip address from mgt0.1dhcp server. ***Result1*** 2. Laptop2 connect to eth4, retrieve ip address from mgt0.1 dhcp server. ***Result2*** | | |
| Expect result | ***Result1:*** laptop1 get ip address successfully assigned by mgt0.1 dhcp server    ***Result2:*** laptop2 get ip address successfully assigned by mgt0.1 dhcp server | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

##### dhcp packet datapath(ap connect to ethx)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_6 | | |
| Priority | High | Automation Flag | No |
| Topology to use | PC1－Switch  |  (eth0)BR(eth3)－AP----STA1 | | |
| Description | Check dhcp packet process for AP with lan ports port trunk mode | | |
| Pre-condition | PC1 connect to switch  AP connect to lan ports eth3  Station1 connect to wifi (AP)  BR lan port eth3 works with trunk mode | | |
| Test procedure | 1. configure mgt0 native-vlan 1 2. configure mgt0.1 vlan 10, mgt0.2 vlan 20 on br200 3. configure lan ports eth3 with trunk mode 4. disable lan port eth3 vlan-id-check 5. check ip address of AP mgt0. ***Result1***   ***Bug 15388: (dhcp discover packet send to BR without vlan-tag, but the offer packet from BR have an native-vlan tag)***  **－Use a laptop connects to lan port eth4, the network card support vlan-tag. Check the laptop ip address.**  **－Use an AP/BR connects to trunk lan port eth4, check mgt0 ip address.**   1. Configure AP mgt0 vlan as BR lan ports native-vlan 10 2. Check ip address of AP mgt0. ***Result2*** 3. STA1 associated an ssid of AP, using upid with vlan 20 4. Check STA1 ip address ***Result3*** 5. Enable vlan-id-check of lan port eth3 6. Re-assicated STA1 to AP ssid ***Result4*** | | |
| Expect result | ***Result1:***  AP mgt0 retrieve ip address from BR mgt0 dhcp server successfully  **Correct result-------the laptop/AP can retrieve the ip address from dhcp server.**  ***Result2:*** AP mgt0 retrieve ip address from BR mgt0.1 dhcp server successfully  ***Result3:***  STA1 can retrieve ip address from BR mgt0.2 dhcp server successfully  ***Result4:*** STA1 can’t get ip address, because the discover packet is dropped by lan port eth3 | | |
| Test result | **BR100: pass**  **BR200: pass** | | |
| Comment |  | | |

##### Raw socket datapath

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_7 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  BR----laptop1 | | |
| Description | Lan ports with same vlan, Check raw socket datapath | | |
| Pre-condition | PC1 connect to switch  AP connect to lan ports ethx  Station1 connect to wifi  BR lan port works with port based mode | | |
| Test procedure | 1. BR lan ports are in same vlan 2. Amrp send raw socket directly to lan port eth1 3. Capture packet on eth1, check src-mac  ***Result1*** 4. Amrp send raw socket directly to lan port eth2 5. Capture packet on eth2, check src-mac  ***Result2*** 6. Amrp send raw socket directly to lan port eth3 7. Capture packet on eth3, check src-mac ***Result3*** 8. Amrp send raw socket directly to lan port eth4 9. Capture packet on eth4, check src-mac ***Result4*** | | |
| Expect result | ***Result1:***  src-mac is eth1  ***Result2:***  src-mac is eth2  ***Result3:*** src-mac is eth3  ***Result4:*** src-mac is eth4 | | |
| Test result | BR100: fail, can’t support, an known bug  BR200: pass | | |
| Comment |  | | |

##### Packet size check (MTU)

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_8 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  BR----laptop1 | | |
| Description | Check traffic when packet size is lesser/greater than MTU | | |
| Pre-condition | PC1 connect to switch  AP connect to lan ports ethx  Station1 connect to wifi  BR lan port works with port based mode | | |
| Test procedure | 1. Laptop1 send a packet to BR cpu(mgt0 ip), packet size is lesser than MTU(1500)   ping 192.168.85.1 -l 1400   1. Check if ping successfully ***Result1*** 2. Laptop1 send a packet to BR cpu(mg0 ip), packet size is greater than MTU(1500)   ping 192.168.85.1 –l 1510   1. Check if ping successfully ***Result2*** | | |
| Expect result | ***Result1:***  ping successfully  ***Result2:*** ping successfully | | |
| Test result | pass | | |
| Comment | Bug15957 | | |

##### Check dhcp pkts forwarding with vlan 1

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_datapath\_9 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  PC++++BR++++AP----station | | |
| Description | Check dhcp pkts forwarding with vlan 1 and check the | | |
| Pre-condition | PC1 connect to switch  AP connect to lan ports eth2  BR lan port eth2 works with trunk mode | | |
| Test procedure | 1. BR create dhcp server mgt0.x with vlan 10,20,30 2. BR lan port eth2 trunk mode, native-vlan 20, allow all 3. AP mgt0 vlan1, native-vlan 30 4. Enable AP mgt0 dhcp client to get ip address ***Result1*** 5. Station bind upid with vlan 10, PC connect with BR access port with vlan 10 6. Check PC ping station ***Result2*** | | |
| Expect result | ***Result1:***  AP mgt0 can get ip address from BR mgt0 vlan 1  ***Result2:***  PC ping station successfully. | | |
| Test result | pass | | |
| Comment |  | | |

### CLI Test Case

#### lan ports mode

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  Laptop1 Laptop2 | | |
| Description | Configure lan ports mode as bridge-access or bridge-802.1q | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR connect to switch | | |
| Test procedure | 1. check default mode for lan ports eth1/eth2/eth3/eth4 ***Result1*** 2. configure eth1 with mode bridge-802.1q   ***CLI:***  ***Interface eth1 mode bridge-802.1q Result2***   1. eth1/eth4 with default vlan 1 2. laptop1 ping laptop2 ***Result3*** 3. no eth1 mode brige-802.1q   ***CLI:***  ***No interface eth1 mode***   1. check interface eth1 mode ***Result4*** 2. laptop1 ping laptop2 ***Result5*** | | |
| Expect result | ***Result1:***  default mode for eth1/eth2/eth3/eth4 is bridge-access  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y port-based 1 -  eth2 D Y port-based 1 -  eth3 U Y port-based 1 -  eth4 U Y port-based 1 -  ***Result2:***  eth1 mode is bridge-802.1q mode  AH-5a3e40#show interface  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:775a:3e40 - U - 1 - hive0 -  Eth0 0019:775a:3e40 wan U - - - - -  Eth1 0019:775a:3e44 bridge U - - - hive0 -  Eth2 0019:775a:3e45 access D - - - hive0 -  Eth3 0019:775a:3e46 access D - - - hive0 -  Eth4 0019:775a:3e47 access U - - - hive0 -  Wifi0 0019:775a:3e48 access D 1 - radio\_ng0 - -  Wifi0.1 0019:775a:3e48 access D 1 - radio\_ng0 hive0 -  AH-5a3e40#show lan port  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y trunk 1 No  eth2 D Y port-based 1 -  eth3 D Y port-based 1 -  eth4 U Y port-based 1 -  ***Result3:*** laptop1 ping laptop2 successfully  ***Result4:*** eth1 mode is change to default mode: backhaul  **-----🡪 can’t use this cli.**  AH-5a3e40#show interface  State=Operational state; Chan=Channel;  Radio=Radio profile; U=up; D=down;  Name MAC addr Mode State Chan VLAN Radio Hive SSID  ------- -------------- -------- ----- ---- ---- ---------- ---------- ---------  Mgt0 0019:775a:3e40 - U - 1 - hive0 -  Eth0 0019:775a:3e40 wan U - - - - -  Eth1 0019:775a:3e44 backhaul U - 1 - hive0 -  Eth2 0019:775a:3e45 access D - - - hive0 -  Eth3 0019:775a:3e46 access D - - - hive0 -  Eth4 0019:775a:3e47 access U - - - hive0 -  Wifi0 0019:775a:3e48 access D 1 - radio\_ng0 - -  Wifi0.1 0019:775a:3e48 access D 1 - radio\_ng0 hive0 -  AH-5a3e40#show lan port  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y trunk 1 No  eth2 D Y port-based 1 -  eth3 D Y port-based 1 -  eth4 U Y port-based 1 -  ***Result5:***  laptop1 ping laptop2 successfully | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

#### lan ports native-vlan for bridge-access

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  Laptop1 Laptop2 | | |
| Description | Configure bridge-access lan ports native-vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR connect to switch | | |
| Test procedure | 1. Lan ports eth1/eth4 work with bridge-access mode 2. Configure mgt0 native-vlan, such as native-vlan 10   ***CLI:***  ***Interface mgt0 native-vlan 10***   1. Show lan port-config, check default vlan for eth1/eth4 ***Result1*** 2. Confingure native-vlan for eth1/eth4 mapped from user-profile attribute.   *CLI:*  *User-profile 10 vlan 10 attribute 10*  *Interface eth1 mode bridge-access user-profile-attribute 10*  *Interface eth4 mode bridge-access user-profile-attribute 10*   1. Show lan port-config, check default vlan for eth1/eth4 ***Result2*** 2. Laptop1 ping laptop2 ***Result3*** | | |
| Expect result | ***Result1:***  the default vlan for access ports should not map from mgt0 native-vlan, so the default vlan still is 1  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y port-based 1 -  eth2 D Y port-based 1 -  eth3 D Y port-based 1 -  eth4 U Y port-based 1 -  ***Result2:*** the default vlan for eth1/eth4 is 1  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y port-based 10 -  eth2 D Y port-based 1 -  eth3 D Y port-based 1 -  eth4 U Y port-based 10 -  ***Result3:***  laptop1 ping lapltop2 successfully | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

#### lan ports native-vlan for bridge-802.1q

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_3 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  Laptop1 Laptop2 | | |
| Description | Configure bridge-802.1q lan ports native-vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR connect to switch | | |
| Test procedure | 1. Lan port eth1/eth4 work with bridge-802.1q mode 2. Configure native-vlan for eth1/eth4 mapped from mgt0 native-vlan, such as native-vlan 10   ***CLI:***  ***Interface mgt0 native-vlan 10***   1. Show lan port-config, check default vlan for eth1/eth4 ***Result1*** 2. Laptop1 ping laptop2  ***Result2*** 3. Confingure native-vlan for eth1/eth4 mapped from eth1/eth4 native-vlan, such as native-vlan is 1 ***Result3***   *CLI:*  *Interface eth1 native-vlan 1*  *Interface eth4 native-vlan 1* | | |
| Expect result | ***Result1:***  the native vlan for eth1/eth4 is 10  ***Result2:***  laptop1 ping laptop2 successfully  ***Result3:*** the native-vlan for eth1/eth4 is 1  *[ all bridge-802.1q interfaces use the same native vlan get from mgt0 native-vlan ]*  *[ per interface native-vlan support: interface ethx native-vlan x ]* | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

#### lan ports allowed-vlan/vlan-id-check for bridge-802.1q

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_4 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  Laptop1 Laptop2 | | |
| Description | Configure bridge-802.1q lan ports allowed-vlan | | |
| Pre-condition | Laptop1 connect to lan port eth1 with trunk mode  Laptop2 connect to lan port eth4 with trunk mode  Laptop3 connect to lan port eth2 with port-based mode configured native-vlan 30  BR connect to switch | | |
| Test procedure | 1. Lan port eth1/eth4 work with bridge-802.1q mode 2. Configure native-vlan for eth1/eth4 mapped from mgt0 native-vlan, such as 10   ***CLI:***  ***Interface mgt0 native-vlan 10***   1. Configure eth3 with trunk mode, and allow-vlan with vlan range ***Result1***   ***Interface eth3 mode bridge-802.1q***  ***Interface eth3 allow-vlan 1-10***  ***Interface eth3 allow-vlan 100***  ***Enable vlan-id-check:* config as auto**   1. Configure eth1/eth4 allow-vlan as auto   ***CLI:***  ***Interface eth1 allow-vlan auto***  ***Interface eth4 allow-vlan auto***   1. Laptop1 ping laptop2 ***Result2*** 2. Laptop1 send packet to laptop2 with vlan tag , matches native-vlan/user-profile vlan ***Result3*** 3. Laptop1 send packet to laptop3 with vlan tag 30 ***Result4***   ***Disable vlan-id-check:***   1. ***CLI:***   ***No interface eth1 allow-vlan auto***   1. Laptop1 send packet with vlan tag 30 ***Result5***   ***Enable vlan-id-check :* configure allow-vlan**   1. Configure allowed-vlan for eth1/eth4 , such as allowed-vlan 20   ***CLI:***  ***Interface eth1 allow-vlan 20***  ***Interface eth4 allow-vlan 20***   1. Show lan port-config/ show lan vlan-table to check vlan configuration ***Result6*** 2. Laptop1 ping laptop2 ***Result7*** 3. Laptop1 send packet to laptop2 with vlan tag which matches native-vlan/allow-vlan, such as vlan tag 20 ***Result8*** 4. Laptop1 send packet to laptop3 with vlan tag not matched native-vlan/allow-vlan, such as vlan tag 30 ***Result9***   ***Disable vlan-id-check:***   1. ***CLI:***   ***Interface eth1 allow-vlan all***   1. Laptop1 send packet to laptop3 with vlan tag not matched native-vlan/allow-vlan, such as vlan tag 30 ***Result10*** | | |
| Expect result | ***Result1:***  allow-vlan set successfully  ***Result2:***  laptop1 ping laptop2 successfully  ***Result3:***  laptop2 can receive packet  ***Result4:*** laptop3 can’t receive packet [not match native-vlan/user-profile vlan, drop packet]  *Result5:* laptop3 can receive packet [disabled vlan-id-check, means allow all]  *Result6:* port-config table shows native-vlan, vlan-table shows allow-vlan list correctly  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y trunk 10 Yes  eth2 D Y port-based 30 -  eth3 D Y port-based 10 -  eth4 U Y trunk 10 Yes  AH-5a3e40#show lan vlan  AH-5a3e40#show lan vlan-table  eth1 eth2 eth3 eth4 VLAN  --------------------------------------------  eth1 eth3 eth4 10  eth2 30  eth1 eth4 20  *Result7:* laptop1 ping laptop2 successfully  *Result8:*  laptop2 can receive packet  *Result9:* laptop3 can’t receive packet [not match native-vlan/allow-vlan, drop packet]  *Result10:*  laptop3 can receive packet [allow all, not do vlan-id-check] | | |
| Test result | BR100: pass  BR200: pass | | |
| Comment |  | | |

#### Lan ports shutdown

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_5 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | Shutdown lan ports eth1/eth2/eth3/eth4, and shutdown interface eth0 | | |
| Pre-condition | Laptop1 connect to lan port eth4  BR connect to switch | | |
| Test procedure | 1. Shutdown interface eth4   ***CLI:***  ***Interface eth4 shutdown***   1. Laptop1 ping laptop2 ***Result1*** 2. No shutdown interface eth1   ***CLI:***  ***No interface eth4 shutdown***   1. Laptop1 ping laptop2 ***Result2*** 2. Execute the CLIs for eth3 ***Result3*** 3. Execute the CLIs for eth2 ***Result3*** 4. Execute the CLIs for eth1 ***Result3*** 5. Shutdown interface eth0 (wan) ***Result4***   *Interface eth0 shutdown*   1. No shutdown interface eth0 *Result5*   *no interface eth0 shutdown* | | |
| Expect result | ***Result1:***  laptop1 ping laptop2 failed  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y trunk 10 Yes  eth2 D Y port-based 30 -  eth3 D Y port-based 10 -  eth4 D  **N** trunk 10 Yes  ***Result2:***  laptop1 ping laptop2 successfully  AH-5a3e40#no interface eth4 shutdown  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 U Y trunk 10 Yes  eth2 D Y port-based 30 -  eth3 D Y port-based 10 -  eth4 U **Y** trunk 10 Yes  ***Result3:***  the CLIs is active for eth2/eth3/eth4  ***Result4:*** shutdown eth0 successfully  ***Result5:*** eth0 is up | | |
| Test result | BR100: pass  BR200: pass, but link and open state in lan port-config table is not updated, it’s by design | | |
| Comment |  | | |

#### Lan ports clear ethx counter

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_6 | | |
| Priority | Low | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | Clear lan ports eth1/eth2/eth3/eth4 counter | | |
| Pre-condition | Laptop1 connect to lan port eth1/eth2/eth3/eth4  BR connect to switch | | |
| Test procedure | 1. Laptop1 connect to lan port eth1 2. Check lan port eth1 counter ***Result1*** 3. Clear lan port eth1 counter   ***clear interface eth1 counter***   1. Check lan port eth1 counter ***Result2*** 2. Check other lan ports status ***Result3*** | | |
| Expect result | ***Result1:***  lan port eth1 has RX/TX packets counter  AH-8d1840#show int eth1  Mode=bridge-802.1q; Mac learning= enabled; Admin state=enabled;  WEB server=disable; NAT support=disable; DHCP client=disable; DHCP server=disable; DNS server=disable;  IP addr=0.0.0.0; Netmask=0.0.0.0;  Internal station traffic state=enabled;  Operational state=up; Duplex=full-duplex; Speed=1000Mbps;  LLDP state=enabled; CDP state=enabled;  Hiveid="xsheng"; Native-vlan=1;  MAC addr=0019:778d:1844; MTU=1500;  Rx packets=13125; errors=0; dropped=0;  Tx packets=13010; errors=0; dropped=0;  Rx bytes=1398411 (1.334 MB); Tx bytes=1751267 (1.670 MB);  ***Result2:***  lan port eth1 RX/TX counters are cleared.  AH-8d1840#show int eth1  Mode=bridge-802.1q; Mac learning= enabled; Admin state=enabled;  WEB server=disable; NAT support=disable; DHCP client=disable; DHCP server=disable; DNS server=disable;  IP addr=0.0.0.0; Netmask=0.0.0.0;  Internal station traffic state=enabled;  Operational state=up; Duplex=full-duplex; Speed=1000Mbps;  LLDP state=enabled; CDP state=enabled;  Hiveid="xsheng"; Native-vlan=1;  MAC addr=0019:778d:1844; MTU=1500;  Rx packets=0; errors=0; dropped=0;  Tx packets=0; errors=0; dropped=0;  Rx bytes=0 (0B); Tx bytes=0 (0 MB);  ***Result3:*** other lan ports is ok | | |
| Test result | pass | | |
| Comment | Bug16590 | | |

#### Lan ports ethx speed

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_CLI\_7 | | |
| Priority | Low | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  |  Laptop1 | | |
| Description | configure lan ports eth1/eth2/eth3/eth4 speed | | |
| Pre-condition | Laptop1 connect to lan port eth1/eth2/eth3/eth4  BR connect to switch | | |
| Test procedure | 1. Laptop1 connect to lan port eth1 2. Check lan port eth1 speed ***Result1*** 3. configure lan port eth1 speed   ***interface eth1 speed 100***   1. Check lan port eth1 speed ***Result2*** 2. Check other lan ports status ***Result3*** | | |
| Expect result | ***Result1:***  lan port eth1 speed is 1000Mbps  AH-8d1840#show int eth3  Mode=bridge-802.1q; Mac learning= enabled; Admin state=enabled;  WEB server=disable; NAT support=disable; DHCP client=disable; DHCP server=disable; DNS server=disable;  IP addr=0.0.0.0; Netmask=0.0.0.0;  Internal station traffic state=enabled;  Operational state=up; Duplex=full-duplex; **Speed=1000Mbps;**  LLDP state=enabled; CDP state=enabled;  Hiveid="xsheng"; Native-vlan=1;  MAC addr=0019:778d:1844; MTU=1500;  Rx packets=17107; errors=0; dropped=0;  Tx packets=17624; errors=0; dropped=0;  Rx bytes=1940466 (1.851 MB); Tx bytes=2468766 (2.354 MB);  ***Result2:***  lan port eth1 speed is 100Mbps  ***Result3:*** this cli also works for other lan ports | | |
| Test result | pass | | |
| Comment | Bug16429 | | |

### Upgrade HiveOS Test Cases

There is so much changed from Congo to Casablanca for the etherent interface configuration CLI,it is better for HM to send a new set of config based on Casablanca HM code.

#### Lan ethx mode commands

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Upgrade\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)  | |  Laptop1 Laptop2 | | |
| Description | Upgrade HiveOS from Congo to Casablanca, check if the commands upgrade for lan ethx mode trunk/port-based | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR connect to switch | | |
| Test procedure | ***Congo:***   1. Configure lan port eth1/eth4 with trunk mode, other lan ports with default config   ***CLI:***  ***Lan eth1 mode trunk***  ***Lan eth4 mode trunk***  ***Lan eth2 mode port-based***  ***Lan eth3 mode port-based***   1. Congo: check lan port-config table ***Result1***   ***Upgrade to Casablanca:***   1. Show running, check if have lan commands ***Result2*** 2. check if lan mode commands is upgrade to interface commands   ***Result3***   1. Check lan port-config table ***Result4*** 2. Laptop1 ping laptop2 ***Result5*** | | |
| Expect result | ***Result1:*** show lan port-config, the mode for lan ports are correct  AH-5a3e40#show lan port-config  Port Link Open Mode Default VLAN VLAN Check  -----------------------------------------------------------------------  eth1 D Y trunk 1 Yes  eth2 D Y port-based 1 -  eth3 D Y port-based 1 -  eth4 U Y trunk 1 Yes  ***Result2:*** show run don’t have lan related clis  ***Result3:*** lan commands is changed to interface commands  ***Casablanca CLI:***  ***Interface eth1 mode bridge-802.1q***  ***Interface eth4 mode bridge-802.1q***  ***Interface eth2 mode bridge-access***  ***Interface eth3 mode bridge-access***  ***Result4:*** show lan port-config table is correct  ***Result5:***  laptop1 ping laptop2 successfully | | |
| Test result | BR100: bug16499 | | |
| Comment |  | | |

#### Vlan-id-check commands

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Upgrade\_2 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)-laptop3  | |  Laptop1 Laptop2 | | |
| Description | Upgrade HiveOS from Congo to Casablanca, check if vlan-id-check upgrades and works | | |
| Pre-condition | Laptop1 connect to lan port eth1 with trunk mode  Laptop2 connect to lan port eth4 with trunk mode  Laptop3 connect to lan port eth2 with port-based mode  BR connect to switch | | |
| Test procedure | ***Congo:***   1. Configure lan port eth1/eth4 with trunk mode, other lan ports with default config   ***CLI:***  ***Lan eth1 mode trunk***  ***Lan eth4 mode trunk***   1. Configure eth1/eth4 with native-vlan 10, allowed-vlan 20,30,40   ***CLI:***  ***interface mgt0/eth1 native-vlan 10***  ***lan eth1/eth4 vlan 20***  ***lan eth1/eth4 vlan 30***  ***lan eth1/eth4 vlan 40***   1. Enable vlan-id-check for lan ports eth1/eth4   ***CLI:***  ***Lan eth1 vlan-id-check***  ***Lan eth4 vlan-id-check***   1. No vlan-id-check for lan ports eth4   ***CLI:***  ***no lan eth4 vlan-id-check***   1. Check port-config table ***Result1***   ***Update to Casablanca:***   1. Show running, check if have lan commands ***Result2*** 2. Check if lan vlan-id-check update to interface commands ***Result3*** 3. Laptop1 send untagged packet to laptop2 ***Result4*** 4. Laptop1 send packet to laptop2 with a tag in [10,20,30.40] ***Result5*** 5. Laptop1 send packet to laptop2 with a tag not match native-vlan and allowed vlan ***Result6*** 6. Laptop3 connec to eth2 which is configured as port-based mode with vlan 50 7. Laptop2 send packet to laptop3 with a tag not match native-vlan and allowed vlan , such as with vlan tag 50 ***Result7*** | | |
| Expect result | ***Result1:***  show lan port-config table, the vlan-id-check is enabled  ***Result2:*** show run don’t have lan related clis  ***Result3:***  the lan command is change to interface commands  ***CLI:***  ***Interface eth1 allow-vlan auto | Interface eth1 allow-vlan 20,30,40***  ***Interface eth4 allow all | no interface eth4 allow-vlan auto***  ***Result4:*** laptop2 can receive the packet  ***Result5:*** laptop2 can receive the packet  ***Result6:***  laptop2 can’t receive the packet  ***Result7:*** laptop3 can receive the packet | | |
| Test result | BR100: bug16127 | | |
| Comment |  | | |

#### HW switch configuration tables

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Upgrade\_3 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)-laptop3  | |  Laptop1 Laptop2 | | |
| Description | Upgrade HiveOS from Congo to Casablanca, check HW switch configuration tabes are still supported, but they are hidden commads | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  Laptop3 connect to lan port eth2  BR connect to switch | | |
| Test procedure | ***Congo:***   1. Configure lan port eth1/eth4 with trunk mode, lan port eth2/eth3 with default config port-based mode 2. Configure lan port eth1/eth4 with native-vlan 10, eth2 native-vlan with 50   ***CLI:***  ***Interface mgt0 native-vlan 10***  ***Interface eth1 native-vlan 10***  ***Lan eth2 vlan 50***   1. Configure lan port eth1/eth4 with allowed-vlan 20, 30,40   ***CLI:***  ***Lan eth1 vlan 20***  ***Lan eth1 vlan 30***  ***Lan eth1 vlan 40***  ***Lan eth4 vlan 20***  ***Lan eth4 vlan 30***  ***Lan eth4 vlan40***   1. Check vlan-table ***Result1***   ***Upgrade to Casablanca:***   1. Show running, check if have lan commands ***Result2*** 2. Show lan mac-table, check if the command support, and the command is hidden ***Result3*** 3. Show lan vlan-table, check if the command is supported, and the command is hidden ***Result4*** 4. Show lan port-config, check if the command is supported, and the command is hidden ***Result5*** | | |
| Expect result | ***Result1:*** show vlan-table, the vlan for eth1/eth4 and eth2 is correct  ***Result2:*** show run don’t have lan related clis  ***Result3:*** the command is hidden, and can still show mac-table, the port field for laptop1/laptop2 should be eth1/eth4  ***Result4:*** the command is hidden, and can still show vlan-table, the native-vlan for eth1/eth4 is 10, allowed-vlan is 20,30,40  ***CLI:***  ***Interface eth1 allow-vlan 20,30,40***  ***Interface eth4 allow-vlan 20,30,40***  ***Interface eth2 native-vlan 50***  ***Result5:*** the command is hidden, and can still show port-config table, the mode for eth1/eth4 is trunk, and for eth2/eth3 is access | | |
| Test result | BR100: bug16501 duplicated with bug16499 | | |
| Comment |  | | |

#### Shutdown lan ports

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Upgrade\_4 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)-laptop3  | |  Laptop1 Laptop2 | | |
| Description | Upgrade HiveOS from Congo to Casablanca, check lan ethx shutdown upgrade to interface ethx shutdown | | |
| Pre-condition | Laptop1 connect to lan port eth1  Laptop2 connect to lan port eth4  BR connect to switch | | |
| Test procedure | ***Congo:***   1. Shutdown lan port eth1   ***CLI:***  ***Lan eth1 shutdown***   1. Laptop1 ping laptop2 ***Result1***   ***Upgrade to Casablanca:***   1. Show running, check if have lan commands ***Result2*** 2. Check if lan ethx shutdown CLI upgrade to interface ethx shutdown ***Result3*** 3. Laptop1 ping laptop2 ***Result4*** 4. No shutdown lan port eth1   ***CLI:***  ***No interface eth1 shutdown***   1. Laptop1 ping laptop2 ***Result5*** | | |
| Expect result | ***Result1:*** laptop1 ping laptop2 failed  ***Result2:*** show run don’t have lan related clis  ***Result3:*** cli upgrade  ***CLI:***  ***Interface eth1 shutdown***  ***Result4:*** laptop1 ping laptop2 failed  ***Result5:*** laptop1 ping laptop2 successfully | | |
| Test result | BR100: pass | | |
| Comment |  | | |

## Stress Test Case

## Duration Test Case

## Performance Test Case

### Lan to lan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Performace\_1 | | |
| Priority | Low | Automation Flag | No |
| Topology to use | BR－laptop2  |  Laptop1 | | |
| Description | Check lan to lan performace | | |
| Pre-condition |  | | |
| Test procedure | Run performace test between lan ports | | |
| Expect result |  | | |
| Test result | BR100: pass, the performance is about 11Mbytes  BR200: pass | | |
| Comment |  | | |

### Lan to wifi

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Performace\_2 | | |
| Priority | Low | Automation Flag | No |
| Topology to use | BR-------S1(wifi)  |  Laptop1 | | |
| Description | Check lan to wifi performance | | |
| Pre-condition |  | | |
| Test procedure | Run performace test between lan and wifi | | |
| Expect result |  | | |
| Test result | BR100: the performace is about 2 Mbytes  BR200: pass | | |
| Comment |  | | |

### Lan to wan

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_LanPorts\_Performace\_2 | | |
| Priority | Low | Automation Flag | No |
| Topology to use | PC－switch  |  BR  |  Laptop1 | | |
| Description | Check lan to wan performace | | |
| Pre-condition |  | | |
| Test procedure | Run performace test between lan and wan   1. Enable eth0 nat ***Result1*** 2. Disable eth0 nat ***Result2*** | | |
| Expect result |  | | |
| Test result | BR100:  Result1: the performance is about 7.5 Mbytes  Result2: the performance is about 8.4 Mbytes  BR200: pass | | |
| Comment |  | | |

## Capacity Test Case

## Compatibility Test Case

## Nagtive Test Case

### BR\_Lanports\_Nagtive\_1

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | BR\_Lanports\_Nagtive\_1 | | |
| Priority | High | Automation Flag | No |
| Topology to use | Switch  |  (eth0)BR(eth1-eth4)－Laptop1(mac-book) | | |
| Description | Check mac learning for lan ports eth1/eth2/eth3/eth4 connect an macbook, can detect vlan tag. | | |
| Pre-condition | Laptop1 connect to BR  Mac-learning feature is enabled for lan ports in BR | | |
| Test procedure | 1. Construct the topology 2. BR eth1/eth4 configure as trunk port ***Result1*** 3. Connect laptop1 to eth1 4. Check if laptop1 can get ip address ***Result2*** 5. Connect laptop1 to eth4 6. Check if laptop1 can get ip adress ***Result3*** | | |
| Expect result | ***Result1:*** it’s ok, eth1/eth4 is trunk port  ***Result2:***laptop1 can get ipaddress  ***Result3:*** laptop1 can get ip address | | |
| Test result | Laptop1 can’t get ip address on BR100, because of bug15388. Trunk port don’t strip native-vlan tag. So laptop1 drop the vlan tagged pkts. | | |
| Comment | BR100 is failed  BR200 is pass | | |

## CLI Management (Automation Status: No)

## GUI Management-HiveManager

<List HM test case or test log>

## GUI Management-HiveUI

<List HiveUI test case or test log>