

TTP Multicast

We will include support for TTP multicast from version 2.8.0. The table number is 40 by default. The pipeline is as follows:

Pipeline

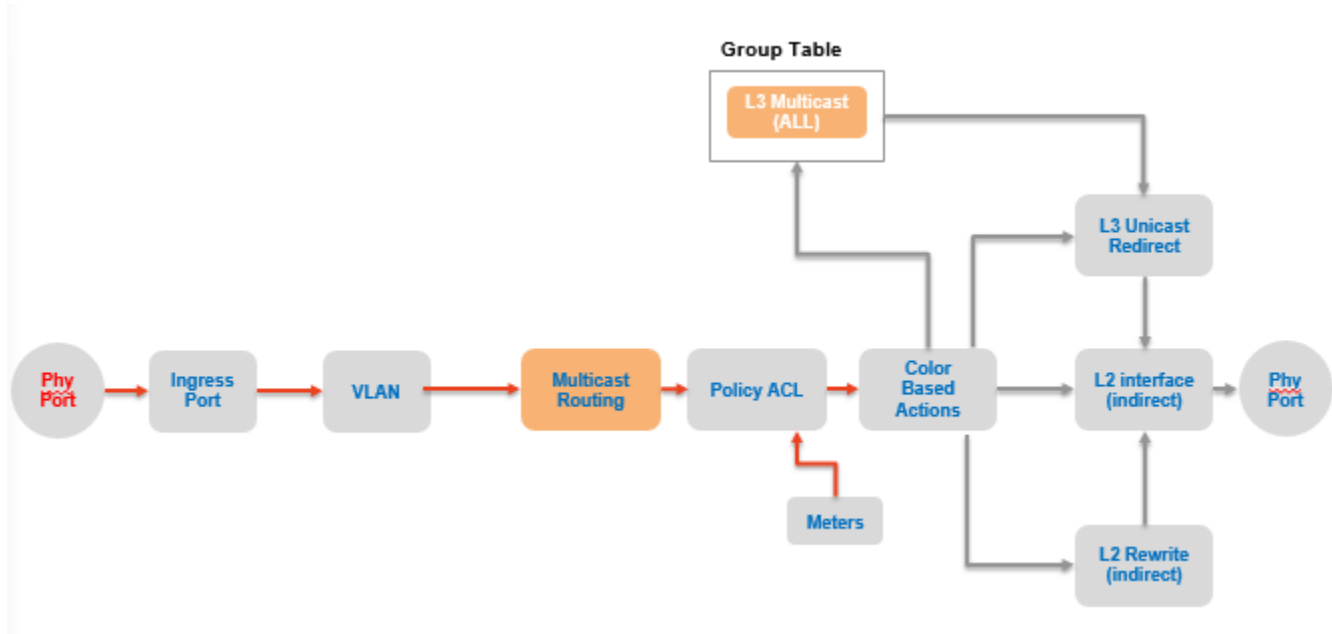


Fig 1:TTP multicast Pipeline

Introduction

The process is different from adding a flow to a unicast routing table. If you add a flow to multicast routing table, you do not need to add the corresponding flows to the VLAN and terminal MAC tables. The switch will check whether the destination MAC of the packet is a multicast address or not. In case its a unicast MAC address, the switch will report an error message. The match fields are Ethernet type, vlan_vid, multicast destination IP address, source IP address. Source ip 0.0.0.0 means (*,g). And (s,g) has the higher priority.

By far, the policy ACL table has ecmp problems in processing the multicast packets coming from multicast routing table.

The multicast routing flow priority is 32768, the IPV4 multicast destination IP address ranges from 224.0.0.0 to 239.255.255.255. The IPV4 destination MAC must begin from 01:00:5e:0..., the IPV6 multicast destination IP address should begin with 0xFF..., the IPV6 multicast destination MAC begins with 33:.....

Table Description

Multicast Routing table

IPV4 multicast

- **table_id:40**
- **priority:32768**
- **match fields:**

eth_type:0x0800

vlan_vid: <vid> 12-bit VLAN id value from header field which Picos support

IPV4_SRC:<ipv4_src>-----IPv4 Unicast Source Address - 32 bits represented as 4 decimal values - nnn.nnn.nnn.nnn

IPV4_DST:<ipv4_mc>-----valid value is from 224.0.0.0 to 239.255.255.255

- **instructions**

goto_table:60 (optional)

write_actions(necessary): L3 Multicast group,DEC_NW_TTL

IPV6 multicast

- **table_id:40**
- **priority:32768**
- **match fields:**

eth_type:0x86dd

vlan_vid: <vid> 12-bit VLAN id value from header field which Picos support

IPV6_SRC:<ipv6_src>-----IPv6 Unicast Source Address - 128 bits represented as 8 groups of hex quartets - nnnn:nnnn:nnnn:nnnn:nnnn:nnnn:nnnn:nnnn

IPV6_DST:<ipv6_mc>-----valid value is ff00:0:0:0:0:0:0:0,mask is ff00:0:0:0:0:0:0:0

- **instructions**

goto_table:60 (optional)

write_actions(necessary): L3 Multicast group,DEC_NW_TTL

default flows in multicast routing table

- **priority:32768**
- **match fields:[]**
- **instructions:**

goto_table:policy table(table_id is 60)

ACL table

match fields:all the match fields the same with tcam in non-TTP mode.

instructions:write_actions,

actions: L2 interface group | L2 Rewrite group | L3 ECMP group | output:controller ,all these actions can not exist at the same time ,only one kind of group exist.

POP_VLAN(optional)

SET_QUEUE(optional)

modify DSCP(optional)

Default flow in ACL table is pass.



I2 multicast and I3 multicast

I2 multicast:the field "dec_nw_ttl" can not work,but in flow,dec_nw_ttl is the necessary.

I3 multicast:del_nw_dec field is necessary and works as well



Multicast table and Acl table can not be matched at the same time.If packets match multicast table,can not match the acl table at the same time unless not match the multicast table.

Groups Description

L2 Interface group

group type is *indirect*,and action only support output port.

L3 Multicast Interface group

group type is *indirect*, actions is modify src mac and vlan, **output** is a "L2 Interface group group".

L3 Multicast group

group type is *all*, bucket is a "**L3 Multicast Interface group**" or "**L2 Interface group**"

Examples

Example 1: IPv4 Multicast (I3 multicast)

(1)configure ttp

```
ovs-vsctl set-ttp-enable true
ovs-vsctl set-ttp-file vz_etech_ttp_v0.3.json
ovs-vsctl show-ttp
sudo /etc/init.d/picos restart
```

(2)configure bridge and port

```
ovs-vsctl add-br br0 -- set bridge br0 datapath_type=pica8
ovs-vsctl add-port br0 ge-1/1/1 vlan_mode=trunk tag=1 -- set interface ge-1/1/1 type=pica8
ovs-vsctl add-port br0 ge-1/1/2 vlan_mode=trunk tag=1 -- set interface ge-1/1/2 type=pica8
ovs-vsctl add-port br0 ge-1/1/3 vlan_mode=trunk tag=1 -- set interface ge-1/1/3 type=pica8
```

(3)add groups

I2 interface group:

```
ovs-ofctl add-group br0 group_id=1,type=indirect,bucket=output:2
ovs-ofctl add-group br0 group_id=2,type=indirect,bucket=output:3
```

I3 multicast interface:

```
ovs-ofctl add-group br0 group_id=3,type=indirect,bucket=set_field:88:88:88:00:00:00->\>d1_src,set_field:200-
\>vlan_vid,group:1
ovs-ofctl add-group br0 group_id=4,type=indirect,bucket=set_field:44:44:44:00:00:00->\>d1_src,set_field:300-
\>vlan_vid,group:2
```

I3 multicast group:

```
ovs-ofctl add-group br0 group_id=5,type=all,bucket=group:3,bucket=group:4
```

(4)add flow

[Here,flows' action goto table:60 is optional.](#)

```
ovs-ofctl add-flow br0 table=40,priority=32768,d1_type=0x0800,d1_vlan=10,nw_src=192.168.1.100,nw_dst=234.
234.245.245,actions=write_actions\ (group:5,dec_nw_ttl\),goto_table:60
```

(5)check the hardware flow table

```
admin@PicOS-OVS$ovs-appctl pica/dump-flows
Multicast Routing Table: (Total 1 flows)
  ID=12 ip,dl_vlan=10,nw_src=192.168.1.100,nw_dst=234.234.245.245, actions:group(id=3,all,n=2,b0(group(id=2,
indirect,n=1,b0(set(dl_src=88:88:88:00:00:00),set(vlan_vid=200),group(id=1,indirect,n=1,b0(live,output:2))))),b1
(live,group(id=6,indirect,n=1,b0(live,set(dl_src=44:44:00:00:00:00),set(vlan_vid=300),group(id=5,indirect,n=1,b0
(live,output:3)))))),goto(Policy ACL Table)
```

(6) send multicast packets

send packets with dst mac "01:00:5E:07:88:88",vlan is 10,dst multicast ipv4 address is 234.234.245.245.

Then ge-1/1/2 will transmit the packets with vlan 200,src mac is 88:88:88:00:00:00,ge-1/1/3 will transmit the packets is src mac is 44:44:44:00:00:00,vlan is 300

Example 2: IPv4 Multicast (I2 multicast)

(1)enable TTP and add bridge and port first.

(2)Add group

I2 interface group:

```
ovs-ofctl add-group br0 group_id=1,type=indirect,bucket=output:2
```

I3 multicast group:

```
ovs-ofctl add-group br0 group_id=3,type=all,bucket=group:1
```

(3)add flow

```
ovs-ofctl add-flow br0 table=40,priority=32768,dl_type=0x0800,dl_vlan=10,nw_src=192.168.1.100,nw_dst=234.
234.245.245,actions=write_actions\ (group:3,dec_nw_ttl\),goto_table:60
```

(4)Check the hardware flow

```
admin@PicOS-OVS$ovs-appctl pica/dump-flows
Multicast Routing Table: (Total 1 flows)
  ID=10 ip,dl_vlan=10,nw_src=192.168.1.100,nw_dst=234.234.245.245, actions:group(id=3,all,n=1,b0(live,group(id=1,
indirect,n=1,b0(live,output:2))))),goto(Policy ACL Table)
```

(5)send packets

send packets with dst mac "01:00:5E:07:88:88",vlan is 10,dst multicast ipv4 address is 234.234.245.245.

Then ge-1/1/2 will transmit the packets which are the same with the packets from ge-1/1/1.

Example 3: IPv6 Multicast(I3 multicast)

(1) enable TTP and add port and bridge.

(2)add group

I2 interface group:

```
ovs-ofctl add-group br0 group_id=1,type=indirect,bucket=output:2
ovs-ofctl add-group br0 group_id=5,type=indirect,bucket=output:3
```

I3 multicast interface group:

```
ovs-ofctl add-group br0 group_id=2,type=indirect,bucket=set_field:88:88:88:00:00:00->\>dl_src,set_field:200-
\>vlan_vid,group:1
ovs-ofctl add-group br0 group_id=6,type=indirect,bucket=set_field:44:44:44:00:00:00->\>dl_src,set_field:300-
\>vlan_vid,group:5
```

I3 multicast group:

```
ovs-ofctl add-group br0 group_id=3,type=all,bucket=group:2,bucket=group:6
```

(3) add flow

```
ovs-ofctl add-flow br0 table=40,priority=32768,dl_type=0x86dd,dl_vlan=10,ipv6_src=2000::1,ipv6_dst=ff00::1,
actions=write_actions\(\group:3,dec_nw_ttl\),goto_table:60
```

(4)Check the hardware flow

```
admin@PicOS-OVS$ovs-appctl pica/dump-flows
Multicast Routing Table: (Total 1 flows)
ID=5040 ipv6,dl_vlan=10,ipv6_src=2000::1,ipv6_dst=ff00::1, actions:group(id=3,all,n=2,b0(live,group(id=2,
indirect,n=1,b0(live,set(dl_src=88:88:88:00:00:00),set(vlan_vid=200),group(id=1,indirect,n=1,b0(live,output:
2))))),b1(live,group(id=6,indirect,n=1,b0(live,set(dl_src=44:44:44:00:00:00),set(vlan_vid=300),group(id=5,
indirect,n=1,b0(live,output:3)))))),goto(Policy ACL Table)
```

(5)send ipv6 mutlicast packets

send ipv6 packets with dst mac 33:33:33:00:00:01,vlan is 10,ipv6 src address is 2000::1,ipv6 dst address is ff00::1

Then ge-1/1/2 will transmit the packets with source mac 88:88:88:00:00:00,vlan is 200,ge-1/1/3 will transmit the packets with dst mac is 33:33:33:00:00:01, src mac is 44:44:44:00:00:00, vlan is 300.

Example 4: IPv6 Multicast (I2 multicast)

(1) enable TTP and add bridge and port

(2)add groups

I2 interface group:

```
ovs-ofctl add-group br0 group_id=1,type=indirect,bucket=output:2
ovs-ofctl add-group br0 group_id=2,type=indirect,bucket=output:3
```

I3 multicast group:

```
ovs-ofctl add-group br0 group_id=3,type=all,bucket=group:1,bucket=group:2
```

(3)add flow

```
ovs-ofctl add-flow br0 table=40,priority=32768,dl_type=0x86dd,dl_vlan=10,ipv6_src=2000::1,ipv6_dst=ff00::1,
actions=write_actions\(\group:3,dec_nw_ttl\),goto_table:60
```

(4)check the hardware flow

```
admin@PicOS-OVS$ovs-appctl pica/dump-flows
Multicast Routing Table: (Total 1 flows)
  ID=1  ipv6,dl_vlan=10,ipv6_src=2000::1,ipv6_dst=ff00::1, actions:group(id=3,all,n=2,b0(group(id=1,indirect,n=1,
b0(output:2))),b1(live,group(id=2,indirect,n=1,b0(live,output:3))))),goto(Policy ACL Table)
```

(5)send ipv6 packets

send ipv6 packets with dst mac 33:33:33:00:00:01,src mac is 22:11:11:11:11:11,vlan is 10,ipv6 src address is 2000::1,ipv6 dst address is ff00::1

Then ge-1/1/2 and ge-1/1/3 will transmit the packets with source mac 22:11:11:11:11:11,vlan is 10.