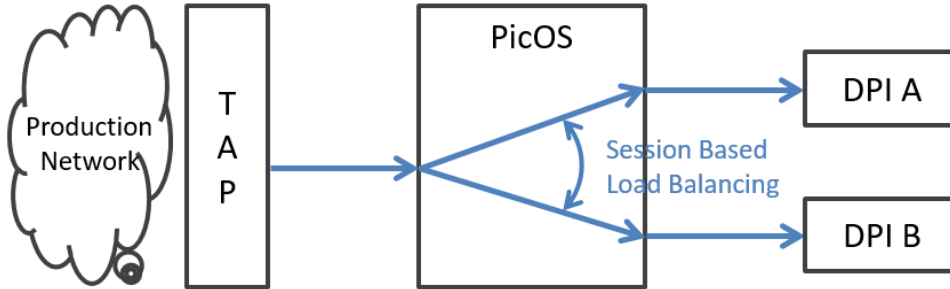


Symmetric Hashing in lag and ecmp

Summary

From PicOS2.9.1, OVS support Symmetric Hashing in LAG interface and ECMP interface.



Traffic from the production network is transferred to DPIs by session based load balancing.

Session based load balancing means same session uplink and downlink packets are transferred to same DPI.

In current image, the packets that are transferred to same DPI have same **IP addresses** and **L4 port numbers** symmetrically with each other.



1, Symmetric hashing is supported on Helix4 , Trident2, Trident2+, Trident3 and Tomahawk platform switches.

2, Symmetric hash key only include: ip address, ip address+I4 port number.

3, If the hash fields include other keys: mac address, vlan, etc. Symmetric hash will be influenced.

Command

Enable/Disable symmetric hash in lag interface

```
ovs-vsctl set-symmetric-hash lag true
ovs-vsctl set-symmetric-hash lag false
```

Except enable lag symmetric hash in global, user also need configure advance field in lag interface.

```
ovs-vsctl -- set Interface ael options:hash-mapping=advance
```

Configure IP addresses symmetrically in lag advance hash mapping fields.

```
ovs-vsctl set-lag-advance-hash-mapping-fields nw_dst nw_src
```

Configure IP addresses and L4 port numbers symmetrically in lag advance hash mapping fields.

```
ovs-vsctl set-lag-advance-hash-mapping-fields nw_dst nw_src port_src port_dst
```

Enable/Disable symmetric has in ecmp interface

```
ovs-vsctl set-symmetric-hash ecmp true
ovs-vsctl set-symmetric-hash ecmp false
```

Configure IP addresses symmetrically in I3 ecmp hash fields.

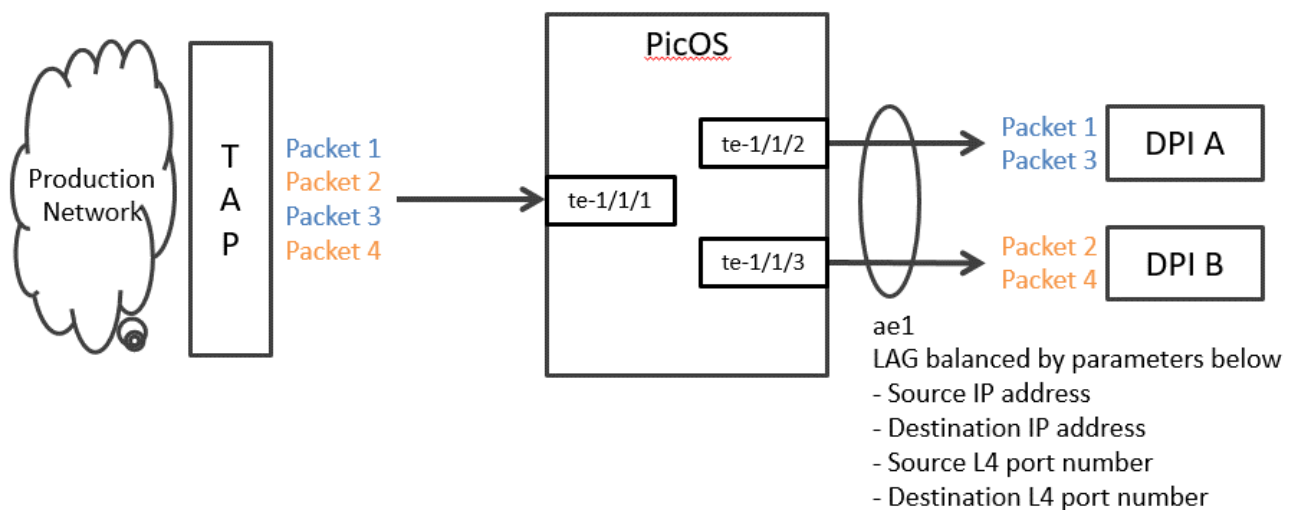
```
ovs-vsctl set-l3-ecmp-hash-fields nw_src nw_dst
```

Configure IP addresses and L4 port numbers symmetrically in l3 ecmp hash fields.

```
ovs-vsctl set-l3-ecmp-hash-fields nw_src nw_dst port_src port_dst
```

Example

1. Configure symmetric in lag interface.



The image of session-based load balancing like below:

Packet X (Src IP, Src Port, Dest IP, Dest Port)

Packet 1 (20.0.0.1, 10001, 10.0.0.1, 80) => output to te-1/1/2 of ae1

Packet 2 (20.0.0.1, 10002, 10.0.0.2, 80) => output to te-1/1/3 of ae1

Packet 3 (10.0.0.1, 80, 20.0.0.1, 10001) => output to te-1/1/2 of ae1

Packet 4 (10.0.0.2, 80, 20.0.0.1, 10002) => output to te-1/1/3 of ae1

-> Packet 1 and Packet 3 are the same session. So the 2 packets are transferred to same LAG member port.

-> Packet 2 and Packet 4 are the same session. So the 2 packets are transferred to same LAG member port.

```
1, Add bridge and ports.
ovs-vsctl add-br br0
ovs-vsctl add-port br0 te-1/1/1 vlan_mode=trunk tag=1
ovs-vsctl add-port br0 te-1/1/2 vlan_mode=trunk tag=1
ovs-vsctl add-port br0 te-1/1/3 vlan_mode=trunk tag=1
ovs-vsctl add-port br0 ael vlan_mode=trunk tag=1 -- set interface ael type=pica8_lag options:members=te-1/1/2,
te-1/1/3

2, Configure advance hash mapping field and apply to ael.
ovs-vsctl -- set Interface ael options:hash-mapping=advance
ovs-vsctl set-lag-advance-hash-mapping-fields nw_dst nw_src port_src port_dst

3, Enable lag symmetric.
ovs-vsctl set-symmetric-hash lag true

4, Add flow entry.
ovs-ofctl add-flow br0 in_port=1,actions=output:1025
```

2. Configure symmetric in ecmp interface.

```
1, Add bridge and ports.
ovs-vsctl add-br br0
ovs-vsctl add-port br0 te-1/1/1 vlan_mode=trunk tag=1
ovs-vsctl add-port br0 te-1/1/2 vlan_mode=trunk tag=1
ovs-vsctl add-port br0 te-1/1/3 vlan_mode=trunk tag=1

2, Configure ecmp group.
ovs-vsctl set-group-ranges ecmp-select-groups=1-10000

3, Configure l3 ecmp hash field.
ovs-vsctl set-l3-ecmp-hash-fields nw_src nw_dst port_src port_dst

4, Enable ecmp symmetric.
ovs-vsctl set-symmetric-hash ecmp true

5, add group and flow.
ovs-ofctl add-group br0 group_id=1,type=select,bucket=set_field:00:00:00:11:11:11->eth_src,set_field:00:00:00:
22:22:22->eth_dst,output:2,bucket=set_field:00:00:00:33:33:33->eth_src,set_field:00:00:00:44:44:44->eth_dst,
output:3
ovs-ofctl add-flow br0 in_port=1,actions=group:1
```