



Implementing LPTS

- [LPTS Overview, on page 1](#)
- [LPTS Policers, on page 1](#)
- [Understanding ACL-based Policers, on page 6](#)

LPTS Overview

Local Packet Transport Services (LPTS) maintains tables describing all packet flows destined for the secure domain router (SDR), making sure that packets are delivered to their intended destinations.

LPTS uses two components to accomplish this task: the port arbitrator and flow managers. The port arbitrator and flow managers are processes that maintain the tables that describe packet flows for a logical router, known as the Internal Forwarding Information Base (IFIB). The IFIB is used to route received packets to the correct Route Processor for processing.

LPTS interfaces internally with all applications that receive packets from outside the router. LPTS functions without any need for customer configuration. However, the policer values can be customized if required. The LPTS show commands are provided that allow customers to monitor the activity and performance of LPTS flow managers and the port arbitrator.

LPTS Policers

Table 1: Feature History Table

| Feature Name | Release Information | Description |
|--|---------------------|---|
| Monitor LPTS Host Path Drops via YANG Data Model | Release 7.3.2 | This feature allows you to use the <code>Cisco-IOS-XR-lpts-pre-ifib-oper.yang</code> data model to monitor the policer action for Local Packet Transport Services (LPTS) flow type for all IOS XR platforms. To access this data model, see the Github repository. |

In Cisco IOS XR, the control packets, which are destined to the Route Processor (RP), are policed using a set of ingress policers in the incoming ports. These policers are programmed statically during bootup by LPTS components. The policers are applied based on the flow type of the incoming control traffic. The flow type is determined by looking at the packet headers. The policer rates for these static ingress policers are defined in a configuration file, which are programmed on the route processor during bootup. You can change the policer values based on the flow types of these set of ingress policers. You are able to configure the rate per policer per node.

Configuration Example

Configure the LPTS policer for the OSPF and BGP flowtypes with the following values:

- ospf unicast default rate 200
- bgp configured rate 200
- bgp default rate 100

```
Router#configure
Router(config)#lpts pifib hardware police
Router(config-pifib-policer-global)#flow ospf unicast default rate 200
Router(config-pifib-policer-global)#flow bgp configured rate 200
Router(config-pifib-policer-global)#flow bgp default rate 100
Router (config-pifib-policer-global)#commit
```

Running Configuration

```
lpts pifib hardware police
flow ospf unicast default rate 200
flow bgp configured rate 200
flow bgp default rate 100
!
```

Verification

```
Router#show run lpts pifib hardware police
lpts pifib hardware police
flow ospf unicast default rate 200
flow bgp configured rate 200
flow bgp default rate 100
```



Note The `show lpts pifib hardware police location 0/RP0/CPU0` command displays pre-Internal Forwarding Information Base (IFIB) information for the designated node.

Configuration Example

Configure the LPTS policer for the OSPF and BGP flow types with the following values:

- ospf unicast default rate 100
- bgp configured rate 300

```
Router#configure
Router(config)#lpts pifib hardware police
Router(config-pifib-policer-per-node)#flow ospf unicast default rate 200
Router(config-pifib-policer-per-node)#flow bgp configured rate 200
```

```
Router(config-pifib-policer-per-node)#flow bgp default rate 100
Router(config-pifib-policer-per-node)#commit
```

Running Configuration

```
lpts pifib hardware police location 0/RP0/CPU0
flow ospf unicast default rate 100
flow bgp configured rate 300
```

Verification

```
Router#show run lpts pifib hardware police
lpts pifib hardware police
flow ospf unicast default rate 100
flow bgp configured rate 300
!
```

Verification

The **show controllers npu stats traps-all instance all location 0/0/CPU0** command displays packets that are locally processed and packets that are dropped by the CPU.

```
Router# show controllers npu stats traps-all instance all location 0/0/CPU0
```

| Trap Type | NPU ID | Trap ID | TrapStats ID | Policer | Packet Accepted | Packet Dropped |
|---|--------|---------|--------------|---------|-----------------|----------------|
| RxTrapMimSaMove (CFM_DOWM_MEP_DMM) | 0 | 6 | 0x6 | 32037 | 0 | 0 |
| RxTrapMimSaUnknown (RCY_CFM_DOWN_MEP_DMM) | 0 | 7 | 0x7 | 32037 | 0 | 0 |
| RxTrapAuthSaLookupFail (IPMC default) | 0 | 8 | 0x8 | 32033 | 0 | 0 |
| RxTrapSaMulticast | 0 | 11 | 0xb | 32018 | 0 | 0 |
| RxTrapArpMyIp | 0 | 13 | 0xd | 32001 | 0 | 0 |
| RxTrapArp | 0 | 14 | 0xe | 32001 | 11 | 0 |
| RxTrapDhcpv4Server | 0 | 18 | 0x12 | 32022 | 0 | 0 |
| RxTrapDhcpv4Client | 0 | 19 | 0x13 | 32022 | 0 | 0 |
| RxTrapDhcpv6Server | 0 | 20 | 0x14 | 32022 | 0 | 0 |
| RxTrapDhcpv6Client | 0 | 21 | 0x15 | 32022 | 0 | 0 |
| RxTrapL2Cache_LACP | 0 | 23 | 0x17 | 32003 | 0 | 0 |
| RxTrapL2Cache_LLDP1 | 0 | 24 | 0x18 | 32004 | 0 | 0 |
| RxTrapL2Cache_LLDP2 | 0 | 25 | 0x19 | 32004 | 1205548 | 0 |
| RxTrapL2Cache_LLDP3 | 0 | 26 | 0x1a | 32004 | 0 | 0 |
| RxTrapL2Cache_ELMI | 0 | 27 | 0x1b | 32005 | 0 | 0 |
| RxTrapL2Cache_BPDU | 0 | 28 | 0x1c | 32027 | 0 | 0 |
| RxTrapL2Cache_BUNDLE_BPDU | 0 | 29 | 0x1d | 32027 | 0 | 0 |
| RxTrapL2Cache_CDP | 0 | 30 | 0x1e | 32002 | 0 | 0 |

| | | | | | | |
|---|---|-----|------|-------|------|---|
| RxTrapHeaderSizeErr | 0 | 32 | 0x20 | 32018 | 0 | 0 |
| RxTrapIpCompMcInvalidIp | 0 | 35 | 0x23 | 32018 | 0 | 0 |
| RxTrapMyMacAndIpDisabled | 0 | 36 | 0x24 | 32018 | 0 | 0 |
| RxTrapMyMacAndMplsDisable | 0 | 37 | 0x25 | 32018 | 0 | 0 |
| RxTrapArpReply | 0 | 38 | 0x26 | 32001 | 2693 | 0 |
| RxTrapFibDrop | 0 | 41 | 0x29 | 32018 | 0 | 0 |
| RxTrapMTU | 0 | 42 | 0x2a | 32020 | 0 | 0 |
| RxTrapMiscDrop | 0 | 43 | 0x2b | 32018 | 0 | 0 |
| RxTrapL2AclDeny | 0 | 44 | 0x2c | 32034 | 0 | 0 |
| Rx_UNKNOWN_PACKET | 0 | 46 | 0x2e | 32018 | 0 | 0 |
| RxTrapL3AclDeny | 0 | 47 | 0x2f | 32034 | 0 | 0 |
| RxTrapOamY1731MplsTp (OAM_SWOFF_DN_CCM) | 0 | 57 | 0x39 | 32029 | 0 | 0 |
| RxTrapOamY1731Pwe (OAM_SWOFF_DN_CCM) | 0 | 58 | 0x3a | 32030 | 0 | 0 |
| RxTrapOamLevel | 0 | 64 | 0x40 | 32023 | 0 | 0 |
| RxTrapRedirectToCpuOamPacket | 0 | 65 | 0x41 | 32025 | 0 | 0 |
| RxTrapOamPassive | 0 | 66 | 0x42 | 32024 | 0 | 0 |
| RxTrap1588 | 0 | 67 | 0x43 | 32038 | 0 | 0 |
| RxTrapExternalLookupError | 0 | 72 | 0x48 | 32018 | 0 | 0 |
| RxTrapArplookupFail | 0 | 73 | 0x49 | 32001 | 0 | 0 |
| RxTrapUcLooseRpfFail | 0 | 84 | 0x54 | 32035 | 0 | 0 |
| RxTrapMplsControlWordTrap | 0 | 88 | 0x58 | 32015 | 0 | 0 |
| RxTrapMplsControlWordDrop | 0 | 89 | 0x59 | 32015 | 0 | 0 |
| RxTrapMplsUnknownLabel | 0 | 90 | 0x5a | 32018 | 0 | 0 |
| RxTrapIpv4VersionError | 0 | 98 | 0x62 | 32018 | 0 | 0 |
| RxTrapIpv4ChecksumError | 0 | 99 | 0x63 | 32018 | 0 | 0 |
| RxTrapIpv4HeaderLengthError | 0 | 100 | 0x64 | 32018 | 0 | 0 |
| RxTrapIpv4TotalLengthError | 0 | 101 | 0x65 | 32018 | 0 | 0 |
| RxTrapIpv4Ttl0 | 0 | 102 | 0x66 | 32008 | 0 | 0 |
| RxTrapIpv4Ttl1 | 0 | 104 | 0x68 | 32008 | 0 | 0 |
| RxTrapIpv4DipZero | 0 | 106 | 0x6a | 32018 | 0 | 0 |
| RxTrapIpv4SipIsMc | 0 | 107 | 0x6b | 32018 | 0 | 0 |
| RxTrapIpv6VersionError | 0 | 109 | 0x6d | 32018 | 0 | 0 |

| | | | | | | |
|--|---|-----|------|-------|-----------|------|
| RxTrapIpv6HopCount0 | 0 | 110 | 0x6e | 32011 | 0 | 0 |
| RxTrapIpv6LoopbackAddress | 0 | 113 | 0x71 | 32018 | 0 | 0 |
| RxTrapIpv6MulticastSource | 0 | 114 | 0x72 | 32018 | 0 | 0 |
| RxTrapIpv6NextHeaderNull | 0 | 115 | 0x73 | 32010 | 0 | 0 |
| RxTrapIpv6Ipv4CompatibleDestination | 0 | 121 | 0x79 | 32018 | 0 | 0 |
| RxTrapMplsTtl1 | 0 | 125 | 0x7d | 32012 | 316278 | 2249 |
| RxTrapUcStrictRpfFail | 0 | 137 | 0x89 | 32035 | 0 | 0 |
| RxTrapMcExplicitRpfFail | 0 | 138 | 0x8a | 32033 | 0 | 0 |
| RxTrapOamp (OAM_BDL_DN_NON_CCM) | 0 | 141 | 0x8d | 32031 | 0 | 0 |
| RxTrapOamEthUpAccelerated (OAM_BDL_UP_NON_CCM) | 0 | 145 | 0x91 | 32032 | 0 | 0 |
| RxTrapReceive | 0 | 150 | 0x96 | 32017 | 125266112 | 0 |
| RxTrapUserDefine_FIB_IPV4_NULL0 | 0 | 151 | 0x97 | 32018 | 0 | 0 |
| RxTrapUserDefine_FIB_IPV6_NULL0 | 0 | 152 | 0x98 | 32018 | 0 | 0 |
| RxTrapUserDefine_FIB_IPV4_GLEAN | 0 | 153 | 0x99 | 32016 | 0 | 0 |
| RxTrapUserDefine_FIB_IPV6_GLEAN | 0 | 154 | 0x9a | 32016 | 0 | 0 |
| RxTrapUserDefine_IPV4_OPTIONS | 0 | 155 | 0x9b | 32006 | 0 | 0 |
| RxTrapUserDefine_IPV4_RSVP_OPTIONS | 0 | 156 | 0x9c | 32007 | 0 | 0 |
| RxTrapUserDefine | 0 | 157 | 0x9d | 32026 | 0 | 0 |
| RxTrapUserDefine_BFD | 0 | 163 | 0xa3 | 32028 | 0 | 0 |
| RxTrapMC | 0 | 181 | 0xb5 | 32033 | 0 | 0 |
| RxNetflowSnoopTrap0 | 0 | 182 | 0xb6 | 32018 | 0 | 0 |
| RxNetflowSnoopTrap1 | 0 | 183 | 0xb7 | 32018 | 0 | 0 |
| RxTrapMimSaMove (CFM_DOWN_MEP_DMM) | 1 | 6 | 0x6 | 32037 | 0 | 0 |
| RxTrapMimSaUnknown (RCY_CFM_DOWN_MEP_DMM) | 1 | 7 | 0x7 | 32037 | 0 | 0 |
| RxTrapAuthSaLookupFail (IPMC default) | 1 | 8 | 0x8 | 32033 | 0 | 0 |
| RxTrapSaMulticast | 1 | 11 | 0xb | 32018 | 0 | 0 |
| RxTrapArpMyIp | 1 | 13 | 0xd | 32001 | 0 | 0 |

Starting Cisco IOS XR Software Release 7.3.2, you can use `Cisco-IOS-XR-lpts-pre-ifib-oper` YANG data model across all IOS XR platforms to retrieve the policer statistics of the flow type. The following example shows the sample RPC request:

```
==== RPC request =====
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter>
      <lpts-pifib xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-lpts-pre-ifib-oper">
```

```

    <nodes>
      <node>
        <node-name>0/0/CPU0</node-name>
        <pifib-hw-flow-policer-stats/>
      </node>
    </nodes>
  </lpts-pifib>
</filter>
</get>
</rpc>
##

```

The following example shows the relevant snippet of the `ICMP-local` flow response to the RPC request:

```

<police-info>
  <flow-type>23</flow-type>
  <flow-name>ICMP-local</flow-name>
  <type>2</type>
  <type-name>Global</type-name>
  <domain-id>0</domain-id>
  <domain-name>default</domain-name>
  <npu-id>255</npu-id>
  <policer-rate>0</policer-rate>
  <burst-size>750</burst-size>
  <accepted>2000</accepted>
  <dropped>1000</dropped>
</police-info>
<police-info>

```

The policer stats of each flow type is the aggregate of all the NPU counters. In the example, the NPU ID of 255 indicates that the value is an aggregate of all NPU stats and provides a simplified view of policer stats per flow type.

Understanding ACL-based Policers

ACL-based LPTS policers are session-based policers that provide secure network access for each session.

Benefits

These are the benefits of ACL-based policer:

- Provides rate limit on incoming packets based on session.
- Modifies policer rates depending on traffic load.
- Blocks entire traffic based on a specific session without impacting other sessions with the same flow.

Restrictions

- It is recommended to have up to 10 prefixes in a single ACL. The ACEs in an ACL should be managed such that there is no overlap of prefixes.
- Up to 50 ACL-based LPTS policers can be configured on a router.
- ACL-based LPTS policers can be configured to an IPv4 LPTS session that has a particular flow type and that matches the default VRF.
- ACL-based LPTS policers can be configured to an IPv4 or IPv6 LPTS session that has a particular flow type and that matches the VRF ID.

Configure IPv4 ACL-based LPTS Policers

This section describes how you can configure IPv4 ACL-based LPTS policers.

Configuration Example

To configure ACL-based LPTS policers for IPv4 sessions, complete the following configurations:

1. Configure an ACL.
2. Configure LPTS policer for a particular IPv4 session matching an ACL and VRF.

Configuration

```

/* Enter the global configuration mode and configure an ACL */
Router# configure
Router(config)# ipv4 access-list ACL1_OSPF
Router(config-ipv4-acl)# 10 permit ipv4 host 192.168.1.5 any
Router(config-ipv4-acl)# commit
Router(config-ipv4-acl)# exit

/* Configure LPTS policer for a particular IPv4 session matching an ACL and VRF. */
Router(config)# lpts pifib hardware police
Router(config-pifib-policer-global)# acl ACL1_OSPF rate 2000 vrf red8
Router(config-if)# commit

```

Verification

Use the following command to display the ACL-based LPTS policer entries attached to matching entries:

```

Router#show lpts pifib hardward entry statistics location 0/RPO/CPU0

```

| Offset | NPU | L3 | VRF id | L4 | Intf | Dest | Pkts/Drops | laddr,Port | raddr,Port | acl_name |
|--------|-----|-----|------------------|---------|------|-------|------------|------------|------------------|----------|
| 37 | 0 | [0] | IPV4 | * | any | any | Local | 0/0 | any,any | |
| 38 | 0 | [1] | IPV4 | * | any | any | Local | 0/0 | any,any | |
| . | . | . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . | . | . |
| 187 | 0 | [0] | IPV4 | * | ICMP | any | Local | 0/0 | any,8 | any,ECHO |
| 188 | 0 | [1] | IPV4 | * | ICMP | any | Local | 0/0 | any,8 | any,ECHO |
| 189 | 0 | [2] | IPV4 | * | ICMP | any | Local | 0/0 | any,8 | any,ECHO |
| 190 | 0 | [3] | IPV4 | * | ICMP | any | Local | 0/0 | any,8 | any,ECHO |
| 8963 | 0 | [0] | IPV4 | default | UDP | OptV2 | Local | 22588/0 | 192.168.10.2,646 | any,any |
| 8964 | 0 | [1] | IPV4 | default | UDP | OptV2 | Local | 22590/0 | 192.168.10.2,646 | any,any |
| 8965 | 0 | [2] | IPV4 | default | UDP | OptV2 | Local | 0/0 | 192.168.10.2,646 | any,any |
| 8966 | 0 | [3] | IPV4 | default | UDP | OptV2 | Local | 0/0 | 192.168.10.2,646 | any,any |
| . | . | . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . | . | . |
| 8279 | 0 | [0] | IPV4 | default | OSPF | OptV2 | Local | 846248/0 | 192.168.10.5,any | any,any |
| 8280 | 0 | [1] | IPV4 | default | OSPF | OptV2 | Local | 962717/0 | 192.168.10.5,any | any,any |
| 8281 | 0 | [3] | IPV4 | default | OSPF | OptV2 | Local | 421169/0 | 192.168.10.5,any | any,any |
| 8281 | 0 | [4] | IPV4 | default | OSPF | OptV2 | Local | 0/0 | 192.168.10.5,any | any,any |
| . | . | . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . | . | . |
| 8339 | 0 | [0] | IPV4 red8 | | OSPF | OptV2 | Local | 0/0 | 192.168.10.5,any | any,any |
| 8340 | 0 | [1] | IPV4 red8 | | OSPF | OptV2 | Local | 0/0 | 192.168.10.5,any | any,any |
| 8341 | 0 | [2] | IPV4 red8 | | OSPF | OptV2 | Local | 11099/0 | 192.168.10.5,any | any,any |
| 8342 | 0 | [1] | IPV4 red8 | | OSPF | OptV2 | Local | 0/0 | 192.168.10.5,any | any,any |

```

.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8363  0 [0] IPV4 red7  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8364  0 [1] IPV4 red7  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8365  0 [2] IPV4 red7  OSPF OptV2 Local 11099/0 192.168.10.5,any any,any  ACL1_OSPF
8366  0 [3] IPV4 red7  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8375  0 [0] IPV4 red6  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8376  0 [1] IPV4 red6  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8377  0 [2] IPV4 red6  OSPF OptV2 Local 11103/0 192.168.10.5,any any,any  ACL1_OSPF
8378  0 [3] IPV4 red6  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8391  0 [0] IPV4 red5  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8392  0 [1] IPV4 red5  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8393  0 [2] IPV4 red5  OSPF OptV2 Local 11104/0 192.168.10.5,any any,any  ACL1_OSPF
8394  0 [3] IPV4 red5  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8411  0 [0] IPV4 red4  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8412  0 [1] IPV4 red4  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8413  0 [2] IPV4 red4  OSPF OptV2 Local 11101/0 192.168.10.5,any any,any  ACL1_OSPF
8414  0 [3] IPV4 red4  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8427  0 [0] IPV4 red3  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8428  0 [1] IPV4 red3  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8429  0 [2] IPV4 red3  OSPF OptV2 Local 11107/0 192.168.10.5,any any,any  ACL1_OSPF
8430  0 [3] IPV4 red3  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8439  0 [0] IPV4 red1  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8440  0 [1] IPV4 red1  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
8441  0 [2] IPV4 red1  OSPF OptV2 Local 11099/0 192.168.10.5,any any,any  ACL1_OSPF
8442  0 [3] IPV4 red1  OSPF OptV2 Local  0/0  192.168.10.5,any any,any  ACL1_OSPF
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
.      .      .      .      .      .      .      .      .      .
8276  0 [1] IPV4 default OSPF OptV2 Local 1752/0 any,any any,any

```

Configure IPv6 ACL-based LPTS Policers

This section describes how you can configure IPv6 ACL-based LPTS policers.

Configuration Example

To configure ACL-based LPTS policers for IPv6 sessions, complete the following configurations:

1. Configure a VRF.
2. Configure an ACL.
3. Configure LPTS policer for a particular IPv6 session matching an ACL and VRF ID.

Configuration

```

/* Enter the global configuration mode and configure an VRF */
Router# configure
Router(config)# vrf red83
Router(config)# commit

/* Configure an ACL */
Router(config)# ipv6 access-list ACL33
Router(config-ipv6-acl)# 10 permit ipv6 host 4000:21::1 host 4000:21::2
Router(config-ipv6-acl)# 60 permit ipv6 host 4000:53::1 host 4000:53::2
Router(config-ipv6-acl)# commit
Router(config-ipv6-acl)# exit

/* Configure LPTS policer for a particular IPv6 session matching an ACL and VRF ID. */
Router(config)# lpts pifib hardware police
Router(config-pifib-policer-global)# acl ACL33 rate 5000 vrf red83 ipv6
Router(config-pifib-policer-global)# commit

```

Verification

Use the following command to display the ACL-based LPTS policer entries attached to matching entries:

```
Router#show lpts pifib hardward entry statistics location 0/RP0/CPU0
```

| Offset | NPU | L3 | VRF id | L4 | Intf | Dest | Pkts/Drops | laddr,Port | raddr,Port | acl name |
|--------|-------|------|--------|-----|------|-------|------------|----------------|------------------|----------|
| 1498 | 0 [0] | IPV6 | red83 | TCP | any | Local | 383/0 | 4000:53::1,179 | 4000:53::1,179 | ACL33 |
| 1499 | 0 [1] | IPV6 | red83 | TCP | any | Local | 0/0 | 4000:53::1,179 | 4000:53::2,43329 | ACL33 |
| 1450 | 0 [2] | IPV6 | red83 | TCP | any | Local | 0/0 | 4000:53::1,179 | 4000:53::2,43329 | ACL33 |
| 1451 | 0 [3] | IPV6 | red83 | TCP | any | Local | 0/0 | 4000:53::1,179 | 4000:53::2,43329 | ACL33 |

