Implementing LPTS

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 `Cisco-IOS-XR-lpts-pre-ifib-oper.yang` 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.

Note You can get the default policer values and the effective current rates of the flow types from the output of the following show command:

```
show lpts pifib hardware police
```

Configuration Example

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

- ospf unicast default rate 3000
- bgp default rate 4000

```
Router#configure
Router(config)#lpts pifib hardware police
Router(config-lpts-policer-global)#flow ospf unicast default rate 3000
Router(config-lpts-policer-global)#flow bgp default rate 4000
Router(config-lpts-policer-global)#commit
```

Running Configuration

```
Router#show running-config lpts
lpts pifib hardware police
 flow ospf unicast default rate 3000
 flow bgp default rate 4000
!
```

Verification

```
Router#show lpts pifib hardware police

-------------------------------------------------------------
                Node 0/RP0/CPU0:
-------------------------------------------------------------
FlowType               Policer Type    Cur. Rate Burst     Accepted     Dropped      npu
---------------------- ------- ------- --------- --------- ------------ ------------ ---------
Fragment               2       np      542       1000      0            0            0
OSPF-mc-known          3       np      1627      1000      0            0            0
OSPF-mc-default        4       np      1084      1000      0            0            0
OSPF-uc-known          5       np      542       1000      0            0            0
OSPF-uc-default        6       np      2878      1000      0            0            0
BFD-default            10      np      8136      1000      0            0            0
BFD-MP-known           11      np      8136      1000      0            0            0
BGP-known              16      np      17000     1000      0            0            0
BGP-cfg-peer           17      np      1627      1000      0            0            0
BGP-default            18      np      3880      1000      0            0            0
```

Configuration Example

Configure the LPTS policer for the OSPF and BGP flow types with the following values on an individual node - 0/0/CPU0:

- ospf unicast default rate 3000
- flow bgp default rate 4000

```
Router#configure
Router(config)#lpts pifib hardware police location 0/0/CPU0
Router(config-lpts-policer-local)#flow ospf unicast default rate 3000
Router(config-lpts-policer-local)#flow bgp default rate 4000
Router(config-lpts-policer-local)#commit
```

Running Configuration

```
Router#show running-config lpts
lpts pifib hardware police location 0/0/CPU0
 flow ospf unicast default rate 3000
 flow bgp default rate 4000
!
```

Verification

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

```
Router#show lpts pifib hardware police location 0/0/CPU0
-------------------------------------------------------------
               Node 0/0/CPU0:
-------------------------------------------------------------
FlowType               Policer Type    Cur. Rate Burst     Accepted     Dropped      npu
---------------------- ------- ------- --------- --------- ------------ ------------ ---------
Fragment               2       np      542       1000      0            0            0
Fragment               2       np      542       1000      0            0            1
OSPF-mc-known          3       np      1627      1000      0            0            0
OSPF-mc-known          3       np      1627      1000      0            0            1
OSPF-mc-default        4       np      1084      1000      0            0            0
OSPF-mc-default        4       np      1084      1000      0            0            1
OSPF-uc-known          5       np      542       1000      0            0            0
OSPF-uc-known          5       np      542       1000      0            0            1
OSPF-uc-default        6       np      2878      1000      0            0            0
OSPF-uc-default        6       np      2878      1000      0            0            1
BFD-default            10      np      8136      1000      0            0            0
BFD-default            10      np      8136      1000      0            0            1
BFD-MP-known           11      np      8136      1000      0            0            0
BFD-MP-known           11      np      8136      1000      0            0            1
BGP-known              16      np      17000     1000      0            0            0
BGP-known              16      np      17000     1000      0            0            1
BGP-cfg-peer           17      np      1627      1000      0            0            0
BGP-cfg-peer           17      np      1627      1000      0            0            1
BGP-default            18      np      3880      1000      0            0            0
BGP-default            18      np      3880      1000      0            0            1
```

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 show 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.

Associated Commands

- **lpts pifib hardware police**
- **flow ospf**
- **flow bgp**
- **show lpts pifib hardware police**

LPTS and NPU Traps

Network Processing Unit (NPU) traps are raised by the routers for inspection. NPU traps are raised in response to the type of packets received by the router and can indicate either exception packets, error packets, or non-LPTS control packets.

- Examples of exception packets include glean adjacency traffic or packets with IPv4 options.
- Examples of error packets include IPv4 packet with bad checksum or IPv6 packets with a hop count of zero.
- Examples of non-LPTS control packets include those packets that do not get processed through LPTS (for example, LACP, LLDP and other L2 control packets).

Each of the NPU traps are policed at a rate that is pre-programmed by the router's system design. Packets are policed per NPU and excess traffic is dropped by the NPU with respect to the system design. Some NPU trap

packets that are allowed by NPU policers are sent to the CPU if they need additional processing. Others that exceed the NPU policer rate are dropped by the NPU.

Verification

Use the command `show controllers npu stats traps-all instance NPU-Number|all location RP|LC` command to check the NPU trap statistics for all the NPUs or per NPU of a router.

For fixed systems, the NPU trap statistics is available for the location 0/RP0/CPU0 and is provided through the command `show controllers npu stats traps-all instance all location 0/RP0/CPU0`. For distributed systems, NPU trap statistics is available for the line card locations and is provided through the command `show controllers npu stats traps-all instance all location 0/1/CPU0`. You can use the command `clear controller npu stats traps-all instance NPU-Number|all location RP|LC`

In the following example:

- **(D)** indicates the trap packets that are dropped in the NPU.
- **(D*)** indicates the trap packets that are dropped in NPU but are available for analysis.
- The **Accepted** count in the output indicates the ones that are available for analysis.

```
RP/0/RP0/CPU0:router#show controllers npu stats traps-all instance all location 0/RP0/CPU0

Trap Type                          NPU  Trap TrapStats  Policer Policer    Packet    Packet

                                   ID   ID   ID                 Rate       Accepted  Dropped
=================================================================================================
ETHERNET_ACL_DROP(D)               0    0    0x0        1       0          0         0
ETHERNET_ACL_FORCE_PUNT(D*)        0    1    0x0        1       0          0         0
ETHERNET_VLAN_MEMBERSHIP(D*)       0    2    0x0        1       0          0         0
ETHERNET_ACCEPTABLE_FORMAT         0    3    0x0        258     100        0         0
UNKNOWN_VLAN_OR_BUNDLE_MEMBER(D*)  0    4    0x0        259     100        0         0
NOT_MY_MAC(D*)                     0    5    0x0        260     100        0         0
ETHERNET_NO_SIP_MAPPING(D*)        0    6    0x0        1       0          0         0
ETHERNET_NO_VNI_MAPPING(D*)        0    7    0x0        1       0          0         0
ETHERNET_NO_VSID_MAPPING(D*)       0    8    0x0        1       0          0         0
ARP                                0    9    0x0        264     542        0         0
ETHERNET_SA_ERROR(D*)              0    11   0x0        266     100        0         0
ETHERNET_DA_ERROR(D*)              0    12   0x0        1       0          0         0
ETHERNET_SA_MULTICAST(D*)          0    13   0x0        268     100        0         0
DHCPV4_SERVER                      0    14   0x0        269     542        0         0
DHCPV4_CLIENT                      0    15   0x0        270     200        0         0
ETHERNET_INGRESS_STP_BLOCK(D*)     0    18   0x0        1       0          0         0
PTP_OVER_ETHERNET                  0    19   0x0        274     4000       0         0
.                                  .    .    .          .       .          .         .
.                                  .    .    .          .       .          .         .
.                                  .    .    .          .       .          .         .
.                                  .    .    .          .       .          .         .
.                                  .    .    .          .       .          .         .
OAMP_BFD_INCORRECT_TTL(D*)         0    157  0x0        412     100        0         0
OAMP_BFD_INVALID_PROTOCOL(D*)      0    158  0x0        413     100        0         0
OAMP_BFD_INVALID_UDP_PORT(D*)      0    159  0x0        414     100        0         0
OAMP_BFD_INCORRECT_VERSION(D*)     0    160  0x0        415     100        0         0
OAMP_BFD_INCORRECT_ADDRESS(D*)     0    161  0x0        416     100        0         0
OAMP_BFD_MISMATCH_DISCR            0    162  0x0        417     500000     0         0
OAMP_BFD_STATE_FLAG_CHANGE         0    163  0x0        418     500000     0         0
OAMP_BFD_SESSION_RECEIVED(D)       0    164  0x0        419     100        0         0
OAMP_PFC_LOOKUP_FAILED(D*)         0    165  0x0        420     100        0         0
```

```
OAMP_PFC_DROP_INVALID_RX(D*)          0     166    0x0          1          0            0          0
APP_SGACL_DROP(D*)                    0     168    0x0          1          0            0          0

Router# show controllers npu stats traps-all instance all location 0/RP/cpu0
Fri Oct 11 05:17:22.720 UTC

Trap Type                                NPU  Trap     TrapStats     Policer              Packet             Packet

                                         ID   ID       ID                                 Accepted           Dropped
=============================================================================================================================
ETHERNET_ACL_DROP(D)                  0    0          0x0           1                    0                  0
ETHERNET_ACL_FORCE_PUNT(D*)           0    1          0x0           1                    0                  0
ETHERNET_VLAN_MEMBERSHIP(D*)          0    2          0x0           1                    0                  0
ETHERNET_ACCEPTABLE_FORMAT                 0    3          0x0           258                      0
 0
UNKNOWN_VLAN_OR_BUNDLE_MEMBER(D*)     0    4          0x0           259                  0                  0
NOT_MY_MAC(D*)                        0    5          0x0           260                  0                  0
ETHERNET_NO_SIP_MAPPING(D*)           0    6          0x0           1                    0                  0
ETHERNET_NO_VNI_MAPPING(D*)           0    7          0x0           1                    0                  0
ETHERNET_NO_VSID_MAPPING(D*)          0    8          0x0           1                    0                  0
ARP                                        0    9          0x0           264                      0
 0
ETHERNET_SA_ERROR(D*)                 0    11         0x0           266                  0                  0
ETHERNET_DA_ERROR(D*)                 0    12         0x0           1                    0                  0
ETHERNET_SA_MULTICAST(D*)             0    13         0x0           268                  0                  0
DHCPV4_SERVER                              0    14         0x0           269                      0
 0
DHCPV4_CLIENT                              0    15         0x0           270                      0
 0
ETHERNET_INGRESS_STP_BLOCK(D*)        0    18         0x0           1                    0                  0
PTP_OVER_ETHERNET                          0    16         0x0           274                      0
 0
.                                          .    .          .             .                        .
 .
.                                          .    .          .             .                        .
 .
.                                          .    .          .             .                        .
 .
.                                          .    .          .             .                        .
 .
.                                          .    .          .             .                        .
 .
OAMP_BFD_INCORRECT_TTL(D*)            0    157        0x0           412                  0                  0
OAMP_BFD_INVALID_PROTOCOL(D*)         0    158        0x0           413                  0                  0
OAMP_BFD_INVALID_UDP_PORT(D*)         0    159        0x0           414                  0                  0
OAMP_BFD_INCORRECT_VERSION(D*)        0    160        0x0           415                  0                  0
OAMP_BFD_INCORRECT_ADDRESS(D*)        0    161        0x0           416                  0                  0
OAMP_BFD_MISMATCH_DISCR                    0    162        0x0           417                      0                  0
OAMP_BFD_STATE_FLAG_CHANGE                 0    163        0x0           418                      0                  0
OAMP_BFD_SESSION_RECEIVED(D*)         0    164        0x0           419                  0                  0
OAMP_PFC_LOOKUP_FAILED(D*)            0    165        0x0           420                  0                  0
OAMP_PFC_DROP_INVALID_RX(D*)          0    166        0x0           1                    0                  0
APP_SGACL_DROP (D*)                   0    168        0x0           1                    0                  0
```

Defining Dynamic LPTS Flow Type

The Dynamic LPTS flow type feature enables you to configure LPTS flow types and also enables you to define the maximum LPTS entries for each flow type in the TCAM. The dynamic LPTS flow type configuration is on per line card basis, hence you can have multiple profiles configured across line cards.

When the router boots, the default LPTS flow types are programmed in the TCAM. For each flow type the maximum flow entries are predefined. Later, at runtime, you have an option to choose the flow type based on network requirements and also configure the maximum flow entry value. The maximum flow entry value of zero denotes that a flow type is not configured.

Note You can get the default maximum flow values for both configurable flow and non-configurable flow from the output of the following show command:

```
show lpts pifib dynamic-flows statistics location <location specification>
```

The list of configurable and non-configurable flow types are listed in below tables. You can also use **show lpts pifib dynamic-flows statistics location** command to view the list of configurable and non-configurable flow types:

Note The sum of maximum LPTS entries configured for all flow types must not exceed 16000 entries per line card.

Configuration Example

In this example you will configure the BGP-known and ISIS-known LPTS flow type in the TCAM and define the maximum flow entries as 1800 and 500 for node location 0/1/CPU0. As the new maximum values are more than the default values, we have to create space in the TCAM by disabling other flow types so that the sum of maximum entries for all flow types per line card does not exceed 8000 entries. Hence RSVP-known flow type is set to zero in our example:

The maximum dynamic scale for any flow type should be configured such that all LPTS entries for that flow type are in hardware. One way to achieve that is to increase the dynamic scale. This may help avoid session flaps for NSR-enabled protocols like BGP and OSPF in case of triggers like RP fail overs.

```
Router#configure
Router(config)#lpts pifib hardware dynamic-flows location 0/1/CPU0
Router(config-pifib-flows-per-node)#flow bgp known max 1800
Router(config-pifib-flows-per-node)#flow rsvp known max 0
Router(config-pifib-flows-per-node)#commit
```

Running Configuration

```
Router#show running-config lpts pifib hardware dynamic-flows location 0/1/CPU0
lpts pifib hardware dynamic-flows location 0/1/CPU0
 flow bgp known max 1800
 flow rsvp known max 0
!
```

Verification

This show command displays dynamic flow statistics. You can see that the flow types BGP-known and ISIS-known are configured in the TCAM with newly configured maximum flow entry value. You can also see that the RSVP-known flow type is disabled:

```
Router#show lpts pifib dynamic-flows statistics location 0/1/CPU0

 Dynamic-flows Statistics:
 -------------------------
 (C - Configurable, T - TRUE, F - FALSE, * - Configured)
 Def_Max  - Default Max Limit
 Conf_Max - Configured Max Limit
 HWCnt    - Hardware Entries Count
 ActLimit - Actual Max Limit
 SWCnt    - Software Entries Count
```

```
P, (+)   - Pending Software Entries

 FLOW-TYPE           C  Def_Max Conf_Max     HWCnt/ActLimit       SWCnt  P
-------------------- -- ------- --------   -------/--------    -------- -
Fragment             F        2       --         2/2                2
OSPF-mc-known        T      600       --         2/600              2
OSPF-mc-default      F        4       --         4/4                4
OSPF-uc-known        T      300       --         1/300              1
OSPF-uc-default      F        0       --         0/0                1  +
BFD-default          F        2       --         2/2                2
BFD-MP-known         T       40       --         1/40               0
BGP-known            T*    2400     1800         6/900              6
BGP-cfg-peer         T      900       --         0/900              0
BGP-default          F        4       --         4/4                4
PIM-mcast-default    F       40       --         0/40               0
PIM-mcast-known      T      300       --         0/300              0
PIM-ucast            F       40       --         2/40               2
IGMP                 T     1200       --         0/1200             0
ICMP-local           F        4       --         4/4                4
ICMP-control         F        5       --         5/5                5
LDP-TCP-known        T      300       --         0/300              0
LDP-TCP-cfg-peer     T      300       --         0/300              0
LDP-TCP-default      F       40       --         0/40               0
LDP-UDP              T      300       --         0/300              0
All-routers          T      300       --         0/300              0
RSVP-default         F        4       --         1/4                1
RSVP-known           T*     300        0         0/0                1  +
SNMP                 T      300       --         8/300              8
SSH-known            T       40       --         0/40               0
SSH-default          T        1       --         1/1                2  +
HTTP-known           T       40       --         0/40               0
SHTTP-known          T       40       --         0/40               0
TELNET-known         T       40       --         0/40               0
TELNET-default       T        1       --         1/1                1
UDP-known            T        0       --         0/0                0
UDP-default          F        2       --         2/2                2
TCP-known            T       40       --         0/40               0
TCP-default          F        2       --         2/2                2
Raw-default          F        2       --         2/2                2
GRE                  F        4       --         0/4                0
VRRP                 T      150       --         0/150              0
DNS                  T       40       --         0/40               0
NTP-known            T       40       --         0/40               0
DHCPv4               T       40       --         0/40               0
DHCPv6               T       40       --         0/40               0
TPA                  T     1000       --         0/1000             0
PM-TWAMP             T       10       --         0/10               0
-----------------------------------------------------
Active TCAM Usage : 13421/16000 [Platform MAX: 16000]
HWCnt/SWCnt       : 65/88
-----------------------------------------------------
```

In the above show command output, the last column **P** specifies the pending software flow entries for the flow type.

User Managed Control Plane and Management Plane ACL

Table 2: Feature History Table

Feature Name	Release Information	Description
Authentication Header (AH) and Encapsulating Security Payload (ESP) Headers Support in User Managed Control Plane and Management Plane ACLs	Release 7.10.1	We've enhanced our traffic security by introducing the Authentication Header (AH) and Encapsulating Security Payload (ESP) IPv6 headers in the IPv6 ACLs. While AH provides data integrity and data origin authentication, ESP is for data confidentiality. You can configure ingress IPv6 ACL extensions for AH and ESP headers to permit or deny packets. These protocols ensure that the sensitive information travelling on the network reaches its destination safely.
User Managed Control Plane and Management Plane ACL	Release 7.3.3 Release 7.5.2	You can create a virtual LPTS interface and apply hybrid ACLs to it for inspecting traffic. This functionality lets you use the hybrid ACLs to filter and customize the control plane and management plane traffic. This feature modifies the following command: • **hw-module profile cef**

On the data plane, all the functions and processes are performed that forward packets from one interface to another. On the control plane, all functions and processes are performed that determine which path to use to forward the packet to the next device. On the management plane, all functions and processes are performed that control and monitor the router. Traditional ACLs, which control and manage data plane traffic, don't allow you monitor control and management plane traffic. With this feature, you can create a virtual (LPTS) interface in the router, which is assigned a hybrid ACL to customize the control plane and management plane traffic, just like the traditional ACL applied on a network interface. You could also configure policer rates in the ACEs of a hybrid ACL with compression level 2 to control and manage the control plane and management plane traffic.

From Release 7.10.1 onwards, you can configure the ACLs to include the Authentication Header (AH) and Encapsulating Security Payload (ESP) headers. The AH and ESP headers are used within IP Security Protocol (IPSec). While AH provides data integrity, ESP provides confidentiality of a packet.

General Guidelines

- You can configure the router to operate in LPTS ACL mode by using the **hw-module profile cef lpts acl** command. To disable the LPTS ACL mode use the **hw-module profile cef lpts acl** command in **no** form.
- The hybrid ACL for control and management plane traffic supports object group match and policer actions. For more information, see Understanding Hybrid ACLs and LPTS Policers, on page 1.
- You must create one LPTS interface for UMPP ACL and include ACEs for control and management plane traffic customization in the same IPv4 or IPv6 ACL.

```
Router (config)# hw-module profile cef lpts acl
Router(config-ipv4-acl)# commit
Router(config-ipv4-acl)# exit
Router(config)# ipv4 access-list test-umpp-v4-filter 10 permit icmp net-group CORP_DC_NETS
 any police 67 pps
Router(config-ipv4-acl)# commit
Router(config-ipv4-acl)# exit
Router(config)# ipv6 access-list test-umpp-v6-filter 10 permit icmpv6 net-group
CORP_DC_NETS any priority Medium
Router(config-ipv6-acl)# commit
Router(config-ipv6-acl)# exit
Router(config)# interface lpts 0
Router(config-if)# ipv4 access-group test-umpp-v4-filter ingress compress level 2
Router(config-if)# ipv6 access-group test-umpp-v6-filter ingress compress level 2
Router(config-if)# commit
Router(config-if)# exit
```

  For detailed information, see Configuring Control Plane and Management Plane Traffic, on page 10.
- The LPTS ACL mode supports only the object group with Level 2 compression.
- You must reboot the router after enabling or the LPTS ACL mode.
- The ACLs for managing control and management plane traffic support configuring policer rate and priority options in the ACE.
- You can enable logging action for the ACLs in this feature.
- By default, the router drops the packets matching deny ACEs. If you must punt such packets, you can use the **icmp-on** option.
- The hybrid ACL for control and management plane traffic does not filter BFD control packets when BFD sessions are hardware offloaded.
- Create the UMPP related object groups before applying the UMPP ingress ACL under the lpts0 interface.
- Reboot the router after you configure your ACLs to include the AH and ESP headers.

Configuring Control Plane and Management Plane Traffic

Use the following configuration to customize control plane and management plane traffic:

```
/* Enable LPTS ACL mode */
Router (config)# hw-module profile cef lpts acl
Router(config-ipv4-acl)# commit
Router(config)# exit

/* Create IPv4 ACL */
```

```
Router(config)# ipv4 access-list test-umpp-v4-filter
Router(config-ipv4-acl)# 10 permit icmp net-group CORP_DC_NETS any police 67 pps
Router(config-ipv4-acl)# 20 permit icmp net-group CORP_OFFICE any priority Medium
Router(config-ipv4-acl)# 30 permit icmp net-group PROD_PRIVATE_V4 any priority High
Router(config-ipv4-acl)# 40 permit icmp net-group PROD_PUBLIC_V4 any police 100 pps
Router(config-ipv4-acl)# 50 permit icmp any any 0
Router(config-ipv4-acl)# 60 permit icmp any any 3
Router(config-ipv4-acl)# priority-timeout 25
Router(config-ipv4-acl)# commit
Router(config-ipv4-acl)# exit

/* Create IPv6 ACL */
Router(config)# ipv6 access-list test-umpp-v6-filter
Router(config-ipv6-acl)# 10 permit icmpv6 net-group CORP_DC_NETS any priority Medium
Router(config-ipv6-acl)# 20 permit icmpv6 net-group CORP_OFFICE any police 67 pps
Router(config-ipv6-acl)# 30 permit icmpv6 net-group PROD_PRIVATE_V6 any priority Low
Router(config-ipv6-acl)# 40 permit icmpv6 net-group PROD_PUBLIC_V6 any police 100 pps
Router(config-ipv6-acl)# 50 permit icmpv6 any any echo
Router(config-ipv6-acl)# 60 permit icmpv6 any any echo-reply
Router(config-ipv4-acl)# priority-timeout 25
Router(config-ipv6-acl)# commit
Router(config-ipv6-acl)# exit

/*Assign the IPv4 and IPv6 ACLs to the virtual LPTS created on enabling the LPTS ACL mode*/
Router(config)# interface lpts 0
Router(config-if)# ipv4 access-group test-umpp-v4-filter ingress compress level 2
Router(config-if)# ipv6 access-group test-umpp-v6-filter ingress compress level 2
Router(config-if)# commit
Router(config-if)# exit

/*Reboot the router*/
```

To disable the LPTS ACL mode, do the following:

```
no hw-module profile cef lpts acl
```

Verification

Use the following commands to verify if the LPTS ACL mode is enabled in the router:

```
Router#show hw-module profile cef
Tue Apr  6 09:06:33.982 UTC
-----------------------------------------------------------------
Knob                          Status          Applied   Action
-----------------------------------------------------------------
CBF                           Unconfigured    N/A       None
BGPLU                         Unconfigured    N/A       None
LPTS ACL                      Configured      Yes       None
Dark Bandwidth                Unconfigured    N/A       None
IP Redirect Punt              Unconfigured    N/A       None
IPv6 Hop-limit Punt           Unconfigured    N/A       None
MPLS Per Path Stats           Unconfigured    N/A       None
Tunnel TTL Decrement          Unconfigured    N/A       None
High-Scale No-LDP-Over-TE     Unconfigured    N/A       None
LPTS Pifib Entry Counters     Unconfigured    N/A       None
```

```
Router#show access-lists test-umpp-v4-filter hardware ingress interface lpts 0 location
0/RP0/CPU0
ipv4 access-list test-umpp-v4-filter
10 permit icmp net-group CORP_DC_NETS any police 67 pps (Accepted: 14 packets, Dropped: 0
packets)
20 permit icmp net-group CORP_OFFICE any priority Medium
30 permit icmp net-group PROD_PRIVATE_V4 any priority High
40 permit icmp net-group PROD_PUBLIC_V4 any police 100 pps (Accepted: 25 packets, Dropped:
```

```
 0 packets)
50 permit icmp any any 0
60 permit icmp any any 3

Router#show access-lists ipv6 test-umpp-v6 hardware ingress interface lpts 0 location
0/RP0/CPU0
ipv6 access-list test-umpp-v6-filter
10 permit icmp net-group CORP_DC_NETS any priority Medium
20 permit icmp net-group CORP_OFFICE any police 67 pps (Accepted: 3 packets, Dropped: 0
packets)
30 permit icmp net-group PROD_PRIVATE_V4 any priority Low
40 permit icmp net-group PROD_PUBLIC_V4 any police 100 pps (Accepted: 35 packets, Dropped:
 0 packets)
50 permit icmp any any echo
60 permit icmp any any echo-reply
```

Configuring ACLs for AH and ESP Headers

Use the following configuration to customize control plane and management plane traffic for AH and ESP headers:

```
/* Enable LPTS ACL mode */
Router (config)# hw-module profile cef lpts acl
Router(config-ipv4-acl)# commit
Router(config)# exit

/* Create IPv6 ACL for AH and ESP header*/
Router(config)# ipv6 access-list ipv6_umpp_access_list
Router(config-ipv6-acl)# 12 permit ahp any any
Router(config-ipv6-acl)# ipv6 access-list ipv6_umpp_access_list
Router(config-ipv6-acl)# 14 permit esp any any
Router(config-ipv6-acl)# commit
Router(config-ipv6-acl)# exit

/*Assign the IPv6 ACLs to the virtual LPTS created on enabling the LPTS ACL mode*/
Router(config-if)# ipv6 access-group ipv6_umpp_access_list ingress compress level 2
Router(config-if)# commit
Router(config-if)# exit

/*Reboot or reload the router*/
Router(config)# reload location 0/0/CPU0
```

Note Ensure that you reboot the router after you configure your ACLs to include the AH and ESP headers.

Running Configuration

```
Router#show running-config
Tue Apr  4 19:34:56.697 UTC
!! Building configuration...
!! IOS XR Configuration 7.10.1.18I
!! Last configuration change at Tue Apr  4 19:21:05 2023 by xyz
!
hostname abc
logging console disable
username xyz
 group root-lr
 group cisco-support
 secret 10
$6$9gZvb/PtyNiT4b/.$2I516Wdlhm7FGa35sKPhmKkGPEibS.chyRzycSzxmFhrR/kdo9JvKCVA7G8gDya1GBtvGDomubxYj1y75gw.g1
!
```

```
line template vty
 exec-timeout 0 0
!
line template test
 exec-timeout 0 0
!
line console
 timeout login response 30
 exec-timeout 0 0
!
line default
 exec-timeout 0 0
 absolute-timeout 0
 session-timeout 0
!
vty-pool default 0 99 line-template vty
call-home
 service active
 contact smart-licensing
 profile CiscoTAC-1
  active
  destination transport-method email disable
  destination transport-method http
 !
!
netconf-yang agent
 ssh
!
ipv4 virtual address 6.7.141.12/16
ipv6 access-list ipv6_umpp_access_list
 10 permit icmpv6 any any packet-too-big
 12 permit ahp any any
 14 permit esp any any
 20 permit icmpv6 any any time-exceeded
 30 permit icmpv6 any any parameter-problem
 40 permit icmpv6 any any echo
 50 permit icmpv6 any any echo-reply
 60 permit icmpv6 any any nd-ns
 70 permit icmpv6 any any nd-na
 80 permit icmpv6 any any
 90 permit udp any any eq bootps
 100 permit udp any any range 33434 33689
 110 permit udp any any eq 1985
 120 permit tcp any eq bgp any
 130 permit tcp any any eq bgp
 140 permit tcp any any eq 57400
 150 permit rsvp any any
 160 permit vrrp any any
 170 permit udp any any range bfd 3785
 180 permit udp any any eq 4784
 190 permit udp any any eq 6784
 200 permit udp any any eq snmp
 210 permit tcp any any eq ssh
 220 permit tcp any any eq telnet
 230 permit udp any eq domain any range 1024 65535
 240 permit udp any any eq ntp
 250 permit udp any any range 1024 1030
 260 permit tcp any any eq 3220
 270 permit udp any eq 3503 any eq 3503
 280 permit tcp any any eq 6666
 290 permit pim any any
 300 permit ospf any any
 320 permit udp any any eq bootpc
 340 permit tcp any any eq www
```

```
 350 permit tcp any any eq https
 360 permit tcp any any range 5900 5910
 370 permit tcp any any range 50000 50100
 380 permit udp any any range 51000 51100
 390 permit udp any any eq 547
 400 permit udp any any eq 546
 402 permit tcp any any eq 6040
 403 permit tcp any any eq 9200
 404 permit tcp any any eq snmp
 405 permit udp any eq 7784 any eq 7784
 406 permit udp any any eq 7784
 407 permit tcp any eq tacacs any
 408 permit tcp any any eq tacacs
 409 permit udp any host ff02::1
 410 permit tcp any any eq 2018
 411 permit tcp any any eq 60100
 430 deny ipv6 any any
!
ipv4 access-list ipv4_umpp_access_list
 10 permit icmp any any fragments
 20 permit icmp any any echo
 30 permit icmp any any echo-reply
 40 permit icmp any any time-exceeded
 50 permit icmp any any unreachable
 60 permit udp any any eq ntp fragment-type first-fragment
 70 permit udp any any fragment-type last-fragment
 80 permit udp any any fragment-type is-fragment
 90 permit udp any any fragments
 100 permit tcp any any eq ssh fragment-type first-fragment
 110 permit tcp any any fragment-type last-fragment
 120 permit tcp any any fragment-type is-fragment
 130 permit udp any any fragments
 140 permit udp any any range 33434 33689
 150 permit rsvp any any
 160 permit tcp any any eq bgp
 170 permit pim any any
 180 permit igmp any any
 190 permit vrrp any any
 200 permit udp any any eq 1985
 210 permit udp any any eq bootps
 220 permit udp any any range bfd 3785
 230 permit udp any any range bfd 4784
 240 permit udp any any range bfd 6784
 250 permit tcp any any eq ssh
 260 permit udp any eq domain any range 1024 65535
 270 permit tcp any any eq telnet
 280 permit udp any eq 3503 any eq 3503
 290 permit tcp any any eq ldp
 300 permit udp any any eq ldp
 310 permit udp any any eq snmp
 320 permit udp any any eq snmptrap
 330 permit tcp any any eq 3220
 340 permit tcp any any eq 6666
 350 permit udp any any eq ntp
 360 permit udp any any range 1024 65535
 370 deny ipv4 any any packet-length range 1 999 fragment-type first-fragment
 380 deny ipv4 any any fragment-offset eq 1
 390 permit ipv4 any any fragment-type first-fragment
 400 permit ipv4 any any fragment-type last-fragment
 420 permit ospf any any
 430 permit icmp any any packet-length range 1000 2999 fragment-type is-fragment
 440 permit udp any any packet-length range 1000 2999 fragment-type is-fragment
 450 permit tcp any any packet-length range 1000 2999 fragment-type is-fragment
 460 permit ipv4 any any packet-length range 1000 2999 fragment-type is-fragment
```

```
 470 permit tcp any any eq 57400
 480 deny ipv4 any any
!
interface MgmtEth0/RP0/CPU0/0
 ipv4 address 6.7.141.11 255.255.0.0
!
interface lpts0
 ipv4 access-group ipv4_umpp_access_list ingress compress level 2
 ipv6 access-group ipv6_umpp_access_list ingress compress level 2
!
interface HundredGigE0/0/0/0
 shutdown
!
interface HundredGigE0/0/0/1
 shutdown
!
interface HundredGigE0/0/0/2
 shutdown
!
<output truncated>
```

Verification

Use the **show access-lists ipv6 hardware interface location** command to verify if the AH and ESP packets are permitted or denied on the router. This example displays the AH and ESP packets that are permitted on the router. The **Accepted** value displays the number of packets permitted.

```
Router# show access-lists ipv6 ipv6_umpp_access_list hardware ingress interface lpts0
location 0/rp0/cpu0
Tue Mar 21 14:24:43.893 UTC
ipv6 access-list ipv6_umpp_access_list
12 permit ahp any any (Accepted: 246524 packets, Dropped: 0 packets)
14 permit esp any any (Accepted: 246524 packets, Dropped: 0 packets)
```