Contents

show ipcp cde

The Inter-Processor Communication Protocol (IPCP) is a light-weight protocol that enables communication between the control plane processor, network processors, and daughter cards. IPCP uses the Classification and Distribution Engine (CDE) which is a full mesh packet switch, and the PCI bus, to send and receive messages.

Sample Output

```
ace3/Admin# show ipcp cde
Rx Msg count                                                  9
Tx Msg count                                                163
Rx byte count                                               432
Tx byte count                                             23552
```

Notes

Field	Description
Rx Msg count	Total messages received by the CDE
Tx Msg count	Total messages transmitted by the CDE
Rx byte count	Total bytes received by the CDE
Tx byte count	Total bytes transmitted by the CDE

show ipcp clients

The Inter-Processor communication protocol (IPCP) is a light-weight protocol that enables communication between the Control Plane Processor (CP), Network Processors (NP) and daughter cards. IPCP uses the Classification and Distribution Engine (CDE), which is a full mesh packet switch, and the PCI bus, to send and receive messages over this path.

The "show ipcp clients" commands shows the SAP processes registered to use IPCP.

The valid SAP IDs within ACE are 49 to 71 and the types are:

- SB_SAP_PEEK_POKE (49)
- SB_SAP_LOOPBACK (50)
- SB_SAP_TEST1 (51)
- SB_SAP_TEST2 (52)
- SB_SAP_TEST_CDE (53)
- SB_SAP_LB_FABRIC (54)
- SB_SAP_HA (55)
- SB_SAP_SYSLOG (56)
- SB_SAP_NTP (57)
- SB_SAP_SME (58)
- SB_SAP_CM (59)
- SB_SAP_FABRIC_TEST (60)
- SB_SAP_DP_DEBUG (61)
- SB_SAP_ICMP_MGR (62)
- SB_SAP_ENCAP (63)
- SB_SAP_IH_FABRIC (64)

- SB_SAP_PROC_INFO (65)
- SB_SAP_SESSION (66)
- SB_SAP_NAT_REAP (67)
- SB_SAP_IFMGR (68)
- SB_SAP_CFGMGR (69)
- SB_SAP_SSL (70)
- SB_SAP_REAP (71)

Sample Output

```
switch/Admin# show ipcp clients

================================
SAP                                    SB_SAP_PEEK_POKE( 49)
uuid                                                      0
pid                                                     735
description                                          insmod

Tx messages                                             262
Rx messages                                             262

Tx bytes                                             561496
Rx bytes                                              11536

Tx dropped messages                                       0
Rx dropped messages                                       0

Tx dropped bytes                                          0
Rx dropped bytes                                          0

Rx Queue Stalls                                           0

Control Q current message count                           0
Control Q max ever                                        1
Control Q max allowed                                   256

Data Q current message count                              0
Data Q max ever                                           0
Data Q max allowed                                      256
```

Notes

The statistics shown are for traffic in the direction of CP to other processors. For each SAP, they are:

Field	Description
Tx dropped messages	SAP message transmitted from CP dropped.
Rx dropped messages	SAP message dropped when receiving due to buffer or memory shortage.
Tx dropped bytes	Number of bytes dropped for the above message count (TX)
Rx dropped bytes	Number of bytes dropped for the above message count (RX)
Rx Queue Stalls	The CP SAP Queue was stuck or stalled meaning all received messages during this time were dropped.

Any high amount of increments of these counters should be of concern and should result in contacting Cisco support.

show ipcp event-history

Prints the internal event log, primarily for the purposes of internal development debugging.

Sample Output

```
ace3/Admin#  show ipcp event-history

1) Event:E_DEBUG, length:72, at 534487 usecs after Fri Dec  7 10:51:26 2007
    [100] (cfgmgr:957) In PCI write, PCI TX Q full path PCI_HI Dst Mod ixp1

 2) Event:E_DEBUG, length:54, at 714216 usecs after Fri Feb  1 14:56:00 2008
    [100] dropping peek/poke pkt reqid 313. no receipient

3) Event:E_DEBUG, length:62, at 409148 usecs after Fri Feb  1 14:56:00 2008
    [100] (cfgmgr:949) Response for mem access request failed -62

4) Event:E_DEBUG, length:91, at 409137 usecs after Fri Feb  1 14:56:00 2008
    [100] ipcp_receive_reply: sleep interrupted by timeout (500) for PID (cfgmgr:949) reqid 313
```

Notes

Not all E_DEBUG conditions are unexpected. For instance, notice the first event which complains that the transmit queue is temporarily blocked because the receiver (xscale) is not dequeuing messages fast enough. This is a temporary condition that will resolve itself.

show ipcp pci

This command displays statistics for the IPCP (inter-processor communication protocol) messages sent over the PCI interface. These messages use a circular buffer of size 64. The output fields "Curr Rx Entry index" and "Curr Tx Entry index" indicate the current location of the active entry in the 64 element ring buffer. This is where the driver is expecting new messages. There are also statistics on Queue alloc/free counters to debug memory leaks.

```
ace3/Admin# show ipcp pci
================================
Source Module    ixp0
================================
Rx Msg count                                          415
Tx Msg count                                          516

Curr Rx Entry index                                    31
Flags                                          IPCP_PCI_FREE
Msg length                                             44

Curr Tx Entry index                                     4
Flags                                          IPCP_PCI_FREE
Msg length                                             62

================================
```

```
Source Module    ixp1
=================================
Rx Msg count                                          414
Tx Msg count                                          515

Curr Rx Entry index                                    30
Flags                                          IPCP_PCI_FREE
Msg length                                             44

Curr Tx Entry index                                     3
Flags                                          IPCP_PCI_FREE
Msg length                                             22

Q alloc count                                         838

Q free count                                          838

peek/poke request alloc count                         827

peek/poke request free count                          827
```

Notes

Field	Description
Rx Msg count	Total number of messages received on the particular IXP
Tx Msg count	Total number of messages transmitted (tx) on the IXP
Curr Rx Entry index	Current shared memory receive (rx) ring index on the IXP
Flags	Flags associated with the current rx entry on the IXP
Msg length	Current rx message length
Curr Tx Entry index	Current shared memory transmit ring index
Flags	Flags associated with current tx entry
Msg length	Current tx message length
Q alloc count	Total number of queue allocations
Q free count	Total number of queue frees
peek/poke request alloc count	Number of peek-poke messages allocated
peek/poke request free count	Number of peek-poke messages freed

show ipcp peek_poke

This command displays information regarding memory read/write requests (aka, peek/poke). IXP memory peek/poke from the Sibyte processor is implemented using IPCP messages over the PCI bus. A special SAP address (0xFF) is reserved for this purpose. This feature uses the application header to specify the opcode (read/write/set) and other parameters. Reliability is ensured by waiting for a response packet from the XSCALE/ME. Therefore the "outstanding peek poke request" will sometimes show peek_poke messages which have not yet been acknowledged by the XSCALE/ME.

Sample Output

```
ace3/Admin# show ipcp peek_poke
Control Q current message count                        0
Control Q max ever                                     2
```

```
Control Q max allowed                                        256
Peek Poke lock owner                                ipcp_pci_rx:742

--------------------------------

Outstanding Peek Poke requests
```

Notes

Field	Description
Control Q current message count	Current number of requests in the peek-poke queue.
Control Q max ever	Largest number of requests in the peek-poke queue.
Control Q max allowed	Maximun allowed number of requests in the peek-poke queue.
Peek Poke lock owner	Current process that is waiting on a response.
Outstanding Peek Poke requests	List of the messages in the queue.

show nat-fabric policies

NAT pools are populated only in one IXP whereas PAT pools are populated in both IXPs. Bitmaps in the output of this command show currently allocated PAT ports and available ports. This can be useful when troubleshooting PAT allocation failures.

Sample Output

```
ace4/Admin# show nat-fabric policies

Nat objects:

NAT object Hash Bucket: 0
        NAT object ID:2 mapped_if:1 policy_id:1 type:DYNAMIC nat_pool_id:2
                Pool ID:2 PAT:1 pool_id:32 mapped_if:1 Ref_count:4 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170 Bitmap-ID:33
                List of NAT object IDs: 7 6 5 2
NAT object Hash Bucket: 4
        NAT object ID:6 mapped_if:1 policy_id:5 type:DYNAMIC nat_pool_id:2
                Pool ID:2 PAT:1 pool_id:32 mapped_if:1 Ref_count:4 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170 Bitmap-ID:33
                List of NAT object IDs: 7 6 5 2
        NAT object ID:8 mapped_if:3 policy_id:7 type:DYNAMIC nat_pool_id:3
                Pool ID:3 PAT:0 pool_id:55 mapped_if:3 Ref_count:1 ixp_binding:in IXP0
                lower:172.16.183.33 upper:172.16.183.45 Bitmap:0x1fff
                List of NAT object IDs: 8
NAT object Hash Bucket: 5
        NAT object ID:5 mapped_if:1 policy_id:4 type:DYNAMIC nat_pool_id:2
                Pool ID:2 PAT:1 pool_id:32 mapped_if:1 Ref_count:4 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170 Bitmap-ID:33
                List of NAT object IDs: 7 6 5 2
NAT object Hash Bucket: 7
        NAT object ID:7 mapped_if:1 policy_id:6 type:DYNAMIC nat_pool_id:2
                Pool ID:2 PAT:1 pool_id:32 mapped_if:1 Ref_count:4 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170 Bitmap-ID:33
                List of NAT object IDs: 7 6 5 2
```

Notes

Field	Description
mapped_if	This is from the output of "show interface internal iftable"
Ref_count	Number of Policy Map Classes this Nat object is configured in
policy_id	Entry created when a NAT action is configured in a Policy Map Class
List of NAT object IDs	NAT objects that have this pool_id in common. PAT: 0 = NAT, 1 = PAT

show netio clients

Displays basic statistics for the Control Plane applications that are transmitting and receiving packets through the NETIO module.

"NETIO" stands for Network Input Output. The "show netio clients" command refers to those ACE processes which receive and transmit packets on the Control Plane (CP) for the ACE itself. Traffic that is destined to the ACE itself arrives at the CP in one of the following ways:

1. Directly from the console connection
2. Directly from the Supervisor Engine Processor (SUP) connection
3. Traffic from the SFI that is forwarded by the CDE in the Data Plane (DP).

This command lists those NETIO clients on the CP which handle the traffic TO and FROM the ACE Module itself. This means traffic to ACE's local Internet Protocol (IP) Interfaces or MAC address. Several NETIO client run on the module, as shown in the sample output and explained below.

Sample Output

```
Active/Admin# show netio clients

--- ARP Mgr ---
Flags = INITIALIZED, HAS_RCVQ, NORMAL-PRIORITY
Rcv Queue Size = 4096, Head = 3, Tail = 3
Rx Pkts = 3, Rx Bytes = 234
Tx Pkts = 8, Tx Bytes = 624
Match Rules:
  Ethertype = 0x806

--- ARP Sync Thread ---
Flags = INITIALIZED, HAS_RCVQ, NORMAL-PRIORITY
Rcv Queue Size = 1024, Head = 0, Tail = 0
Rx Pkts = 0, Rx Bytes = 0
Tx Pkts = 0, Tx Bytes = 0
Match Rules:

--- Health Mon ---
Flags = INITIALIZED, HAS_RCVQ, NORMAL-PRIORITY
Rcv Queue Size = 24576, Head = 0, Tail = 0
Rx Pkts = 0, Rx Bytes = 0
Tx Pkts = 0, Tx Bytes = 0
Match Rules:
  IP Protocol = 1
  ICMP ID = 2

--- ICMP Manager ---
Flags = INITIALIZED, HAS_RCVQ, ICMP_MGR, NORMAL-PRIORITY
Rcv Queue Size = 4096, Head = 0, Tail = 0
Rx Pkts = 0, Rx Bytes = 0
```

```
Tx Pkts = 0, Tx Bytes = 0
Match Rules:
  IP Protocol = 1
  ICMP ID = 1

--- Interface Manager ---
Flags = INITIALIZED, NORMAL-PRIORITY
Rcv Queue Size = 0, Head = 0, Tail = 0
Rx Pkts = 0, Rx Bytes = 0
Tx Pkts = 6, Tx Bytes = 576
Match Rules:

--- BPDU Handler ---
Flags = INITIALIZED, HAS_RCVQ, NORMAL-PRIORITY
Rcv Queue Size = 1024, Head = 0, Tail = 0
Rx Pkts = 0, Rx Bytes = 0
Tx Pkts = 0, Tx Bytes = 0
Match Rules:
  Ethertype = 0x10b
    SNAP Header Required
```

Notes

The NETIO clients which run on the CP are as follows:

- Address Resolution Protocol (ARP) Manager ? This task is responsible for the ACE ARP control across all configured contexts. This includes sending ARP Requests, replying to ARP Requests and processing ARP Replies.
- Internet Control Message Protocol (ICMP) Manager ? This task is responsible for the ACE ICMP control across all configured contexts. This includes generating ICMP packets, replying to ICMP packets and acting upon ICMP packets when needed (for example and ICMP Redirect).
- ARP Synchronization Manager ? This task runs in conjunction with the ARP Manager to coordinate ARP packets transmitted by the ACE.
- Health Monitoring (HM) Manager ? This task is responsible for all configured PROBEs on the ACE across all configured contexts. It manages the PROBE connection transmitting and receiving the PROBE traffic. It also publishes the state of the PROBE so that rserver and other configuration attributes which rely on configured PROBEs are properly marked UP/DOWN.
- Bridge Protocol Data Unit (BPDU) Handler ? This tasks is responsible for running Spanning Tree Protocol (STP) across all configured contexts. This includes the transmission and receiving of BPDU and populating the VLAN bridge and state tables.
- Interface Manager ? This task is responsible for the interface statistics across all configured contexts for the VLANs.
- System Logging (SYSLOG) Manager ? This task is responsible for the sending of SYSLOG messages across all configured contexts for any defined logging hosts.

The output format for each process is the same. Here is an example from "show netio clients" when debugging an HM problem with ICMP PROBEs:

```
--- Health Mon ---
Flags = INITIALIZED, HAS_RCVQ, NORMAL-PRIORITY
Rcv Queue Size = 24576, Head = 2253, Tail = 2253
Rx Pkts = 813261, Rx Bytes = 74820012
Tx Pkts = 813278, Tx Bytes = 74822416
```

The fields in the output are:

Field	Description

Rx Pkts	Number of packets received to the ACE by this NETIO Client.
Rx Bytes	Number of bytes received in these packets.
Tx Pkts	Number of packets transmitted by the ACE for this NETIO Client.
Tx Bytes	Number of packets in these packets.

A few more points to note:

- In general "Head" and "Tail" should be the same value; if they are not consistently the same it would indicate that the queue is stuck and packets are not being processed. This is cause for concern and should be escalated to Cisco TAC. If in our case the HM NETIO queue was stuck, we would see PROBEs go down.
- A difference in the Receive and Transmitted packets can indicate an error. In this case the difference between "Rx Pkts" and "Tx Pkts" was 17, which corresponded to ICMP Errors detected by the ACE so the ICMP PROBE responses were being dropped.

show netio event-history

Displays a historic log of the most recent Control Plane network I/O debug messages.

Sample Output

```
1) Event:E_DEBUG, length:70, at 187619 usecs after Tue Jan 29 22:17:52 2008
    [101] ed_transmit_pkt: MTU-IMPH failed, interface 1 is being modified

2) Event:E_DEBUG, length:70, at 54065 usecs after Tue Jan 29 22:17:45 2008
    [101] ed_transmit_pkt: MTU-IMPH failed, interface 1 is being modified

3) Event:E_DEBUG, length:70, at 767589 usecs after Tue Jan 29 22:17:39 2008
    [101] ed_transmit_pkt: MTU-IMPH failed, interface 1 is being modified
```

show netio stats

Displays detailed counters for various Control Plane Network Input/Output (NETIO) events.

The "show netio stats" command is one of several that can be used to verify that traffic is going to the ACE. Traffic that is destined to the ACE itself arrives at the Control Plane (CP) in one of the following ways:

1. Directly from the console connection
2. Directly from the Supervisor Engine Processor (SUP) connection
3. Traffic from the SFI that is forwarded by the CDE in the Data Plane (DP)

Sample Output

```
High Priority (Control)              Normal Priority (Data)
-----------------------              ----------------------
Net Rx Packets         : 119         Net Rx Packets         : 80005
Net Rx Bytes           : 9280        Net Rx Bytes           : 6250441
Net Rx Unsupported L2  : 0           Net Rx Unsupported L2  : 0
Net Rx Lock Errors     : 0           Net Rx Lock Errors     : 0
Net Rx Interface Miss  : 115         Net Rx Interface Miss  : 79939
Net Rx No Arp Client   : 0           Net Rx No Arp Client   : 0
```

```
Net Rx Alias Drops    : 0           Net Rx Alias Drops    : 0
Net Rx Repl. Errors   : 0           Net Rx Repl. Errors   : 0
Net Rx Repl. If Err   : 0           Net Rx Repl. If Errs  : 0
Net Rx Internal Errs  : 0           Net Rx Internal Errs  : 0

Net Tx Packets        : 0           Net Tx Packets        : 84
Net Tx Bytes          : 0           Net Tx Bytes          : 17029
Net Tx Lock Errors    : 0           Net Tx Lock Errors    : 0
Net Tx Bad Context ID : 0           Net Tx Bad Context ID : 0
Net Tx No Route Found : 0           Net Tx No Route Found : 0
Net Tx No Adjacency   : 0           Net Tx No Adjacency   : 0
Net Tx Invalid If ID  : 0           Net Tx Invalid If ID  : 0
Net Tx If Down        : 0           Net Tx If Down        : 0
Net Tx No Src Addr    : 0           Net Tx No Src Addr    : 0
Net Tx No Encap       : 0           Net Tx No Encap       : 0
Net Tx FIFO Errors    : 0           Net Tx Fifo Errors    : 0
Net Tx No VMAC Errors : 0           Net Tx No VMAC Errors : 0

IPC Tx Packets        : 55          IPC Tx Packets        : 0
IPC Tx Bytes          : 3272        IPC Tx Bytes          : 0
IPC Tx Fifo Errors    : 0           IPC Tx Fifo Errors    : 0

Client Rx Queue Full  : 0           Client Rx Queue Full  : 0
```

Notes

Field	Description
Net Rx Packets	Number of packets received from the FIFO channel.
Net Rx Bytes	Number of bytes received from the FIFO channel.
Net Rx Unsupported L2	Number of packets received (and dropped) with an unsupported L2 encapsulation.
Net Rx Lock Errors	Number of times a received packet was dropped because a data structure (interface or encap entry) was under modification.
Net Rx Interface Miss	Number of packets received on an interface (vlan) which is either invalid or DOWN.
Net Rx No Arp Client	Number of ARP packets received when no ARP application was registered to receive the packet. Could indicate a potential problem with the ARP module.
Net Rx Repl. Errors	Number of buffer allocation failures when trying to replicate a broadcast packet on a shared interface for multiple contexts.
Net Rx Repl. If Err	Number of times a broadcast packet on a shared interface could not be replicated for multiple contexts because one interface was invalid or DOWN.
Net Rx Internal Errs	Number of unexpected internal errors processing received packets.
Net Tx Packets	Number of packets transmitted on the FIFO channel.
Net Tx Bytes	Number of bytes transmitted on the FIFO channel.
Net Tx Lock Errors	Number of times a transmitted packet was dropped because a data structure (interface or encap entry) was under modification.
Net Tx Bad Context ID	Number of times an invalid context was used for an outgoing route lookup.
Net Tx No Route Found	Number of packets which could not be transmitted because no suitable route was found.

Net Tx No Adjacency	Number of packets which could not be transmitted because no suitable ARP (adjacency) entry was found.
Net Tx Invalid If ID	Number of packets which could not be transmitted because the outgoing interface for the packet is invalid.
Net Tx If Down	Number of packets which could not be transmitted because the outgoing interface for the packet is DOWN.
Net Tx No Src Addr	Number of packets which could not be transmitted because the outgoing interface had no IP address configured.
Net Tx No Encap	Number of packets which could not be transmitted because there was an error constructing the L2 layer for the packet. Most likely cause is the encap (ARP) table was being modified at the time.
Net Tx FIFO Errors	Number of packets which could not be transmitted because of FIFO driver errors. Check the FIFO stats for more details.
Net Tx No VMAC Errors	Number of packets which could not be transmitted because the L2 layer required a virtual MAC address and none was available on the outgoing interface.

show np 1 access-list resource

Available from the Admin context, this command shows the memory allocation and limits for the different nodes in the ACL merge tree. The nodes are Compressed, Uncompressed, Leaf Head, Leaf Parameter and Policy action nodes. This command shows the maximum limit for each type of node except the Policy Action nodes.

Sample Output

```
ACE30001/Admin#  show np 1 access-list resource

ACL Tree Statistics for Context ID: Admin
=========================================
ACL memory max-limit: None
ACL memory guarantee: 0.00 %
MTrie nodes(used/guaranteed/max-limit):
      51 / 0 / 262143 (compressed)
      4 / 0 / 19999 (uncompressed)
Leaf Head nodes (used/guaranteed/max-limit):
      41 / 0 / 204799
Leaf Parameter nodes (used/guaranteed/max-limit):
      604 / 0 / 409600
Policy action nodes used: 154
memory consumed: 26608 bytes resource-limited 4928 bytes other 31536 bytes total.
min-guarantee: 0 bytes total.
max-limit: 70844416 bytes total, 0 % consumed.

ACL Tree Statistics for the linecard
====================================
MTrie nodes(used): 1835 (compressed) 1285 (uncompressed)
           (shared): 170343 (compressed) 13002 (uncompressed)
Leaf Head nodes (used/shared): 40723 / 123613
Leaf Parameter nodes (used/shared): 207208 / 130346
Policy action nodes (used/shared): 202396 / 2403
Memory consumed 8373680 bytes resource-limited 6476672 bytes other 14850352 bytes total.
```

Notes

In the sample output, notice that the ACE has 262143 compressed nodes, 19999 uncompressed nodes, 204799 Leaf Head Nodes, 409600 Leaf Parameter Nodes and 204800 Policy Action nodes. The current system-wide Policy Action node usage, 202396, is close to the maximum system limit, which is 204800.

When you configure ACL resources, such as min 10% and max *equal to min*, ACE guarantees 10% of each nodes, except action nodes, to that resource class. What that means is ACE configurations for that resource class need to be accommodated with 10% of each node. If any of the nodes consumption goes beyond the 10% limit, ACL resource allocation would fail even though the other nodes usages are well below 10%.

ACL node usage depends on the nature of the configuration. However, it does not depend directly on the number of lines in the config; that is, it doesn't increase linearly with the number of lines of configurations. Node allocation for a given configuration is handled by the ACL compiler using a complex data structure, and it is therefore difficult to calculate the exact node usage for a particular configuration before it is applied.

Thus, to determine whether resources are sufficiently allocated, it is recommended that you:

1. Apply the configuration on the ACE.
2. Find the maximum used node, in terms of percentage, for the applied configuration from the **show np 1 access-list resource** command.
3. While configuring the ACL resources, make sure the max resource percentage is above the percentage calculated for each node type by dividing the used nodes value by the maximum number of nodes and multiplying the result by 100.

In the sample, we can see that the Leaf Parameter Nodes are the most used nodes. We can calculate the Leaf Parameter node percentage by the following method. (Please note that Action Nodes are not part of the resource calculation. However, there is a system-wide limit for action nodes of 204800 and the current system-wide usage can be seen from Admin context using **show np 1 resource** command.)

The number of Leaf Parameter nodes used is 73104, while the maximum limit is 409600.

Thus, the percentage of Leaf Parameter nodes is: 73104 / 409600 x 100 = 17.8%

So for the ACL resource configuration with this configuration example, the ACE administrator needs to ensure that the MAX limit for the ACL resource is at least 17.8%.

Similarly we can calculate the percentage number used for the other ACL nodes.

Further notes:

1. The section "ACL Tree Statistics for the linecard" is present for the command in Admin context but not in user-defined contexts.
2. There are two forms of the command **show np 1 access-list resources** and **show np 2 access-list resources**. However, the output for each should be the same. If it isn't, it warrants further investigation.

For more information, see Troubleshooting Access Control Lists

show np 1 adj

This command shows the adjacency database (that is, the encaps database) for the np. Note that the **show np 1 adj** command is context-sensitive, and shows different information for different contexts.

The purpose of the encaps database is to keep a copy of all the L2 header information required to send a packet to the likely destinations. ACE can do a single look up for the L2 data necessary to "encapsulate" the L3 message for transmission. The lookup is efficient, as it is based on the L3 (IPv4) header.

There is also a "reverse encaps" database, which does a fast lookup based on the MAC address of the incoming packet.

Sample Output

For the admin context, sample output is:

```
switch/Admin# sho np 1 adj
   id   S:Ver flag  imph 1  imph 2           DstMac           SrcMac  MTU ifid
-------------------------------------------------------------------------------
    1 1:0       1   8001e       0 ffff:ffff:ffff 000a:b866:74f7   1500     2
    2 1:0       1   8000a       0 0204:0602:f2d1 000b:fcfe:1b02   1500     5
    3 1:0       1   80014       0 ffff:ffff:ffff 000b:fcfe:1b02   1500     3
    4 1:0       1   80002       0 ffff:ffff:ffff 000b:fcfe:1b02   1500     4
    5 1:0       1   80014       0 00e0:8124:8085 000b:fcfe:1b02   1500     3
    6 1:0       1   80014       0 00e0:8124:7b8d 000b:fcfe:1b02   1500     3
   16 1:0       1   80014       0 0010:585d:314c 000b:fcfe:1b02   1500     3
   17 1:0       1   8000a       0 0001:9670:abe0 000b:fcfe:1b02   1500     5
   18 1:0       1   8000a       0 0204:0602:f2d1 000b:fcfe:1b02   1500     5
   19 1:0       1   8001e       0 0018:b9a6:9079 000a:b866:74f7   1500     2
   20 3:0       1   80002       0 0018:b9a6:9079 000b:fcfe:1b02   1500     4
   21 1:0       1   80002       0 00e0:8120:7267 000b:fcfe:1b02   1500     4
total valid encap entries = 12
total invalid encap entries = 32755
```

For a particular context, sample output appears as follows.

```
switch/c1# show np 1 adj
   id   S:Ver flag  imph 1  imph 2           DstMac           SrcMac  MTU ifid
-------------------------------------------------------------------------------
    2 1:0       1   8000a       0 0204:0602:f2d1 000b:fcfe:1b02   1500     5
    3 1:0       1   80014       0 ffff:ffff:ffff 000b:fcfe:1b02   1500     3
    4 1:0       1   80002       0 ffff:ffff:ffff 000b:fcfe:1b02   1500     4
    5 1:0       1   80014       0 00e0:8124:8085 000b:fcfe:1b02   1500     3
    6 1:0       1   80014       0 00e0:8124:7b8d 000b:fcfe:1b02   1500     3
   16 1:0       1   80014       0 0010:585d:314c 000b:fcfe:1b02   1500     3
   17 1:0       1   8000a       0 0001:9670:abe0 000b:fcfe:1b02   1500     5
   18 1:0       1   8000a       0 0204:0602:f2d1 000b:fcfe:1b02   1500     5
   20 3:0       1   80002       0 0018:b9a6:9079 000b:fcfe:1b02   1500     4
   21 1:0       1   80002       0 00e0:8120:7267 000b:fcfe:1b02   1500     4
total valid encap entries = 10
total invalid encap entries = 32757
```

Notes

Field	Description
id	

	The index into the encaps database, which is kept in DRAM. (That is, there is one encaps database per IXP, and they should both contain the same information.)
S	Sequence number.
Ver	Version of encaps.
flag	Shows various things about the encaps entry. A flag value of "1" shows the encaps is valid.
imph1	First 32 bits of the inter module protocol header (IMPH).
imph2	Second 32 bits. Note that the IMPH headers are not parsed with the standard protocol.
Dest MAC and Source MAC	The destination and source MAC addresses. A different source MAC is used when we use the Burned In Address (BIA) for a MAC, when a shared VLAN MAC (0012:43 ...) and when we use an alias MAC (e.g., 000b:fcfe:1b02). There is a different encaps entry for each destination on a VLAN, even if the same source MAC is used.
ifid	The interface VLAN on which this header will be used. Note that the "invalid encaps" plus the valid (listed) encaps), always add up to 0x7fff. The entire array is there, by definition; it's just the valid entries that are of interest.

show np 1 cpu

This command displays process CPU Information. It has two forms, **show np 1 cpu** and **show np 2 cpu**.

Sample Output

```
switch/Admin# show np 2 cpu

================================
Per-Thread Information
================================
     pid tid name               prio STATE       Blocked
       1   1 proc/boot/procnto    0f READY
       1   2 proc/boot/procnto  255r RECEIVE     1
       1   3 proc/boot/procnto  255r RECEIVE     1
       1   4 proc/boot/procnto   11r RECEIVE     1
       1   5 proc/boot/procnto   10r RECEIVE     1
       1   6 proc/boot/procnto   10r RUNNING
       1   7 proc/boot/procnto   10r RECEIVE     1
       1   8 proc/boot/procnto   10r RECEIVE     1
       1   9 proc/boot/procnto   10r RECEIVE     1
       2   1 vc-ser8250-ixp2400  10r RECEIVE     1
       3   1 proc/boot/devf-ram  10r SIGWAITINFO
       3   2 proc/boot/devf-ram  10r RECEIVE     1
       3   3 proc/boot/devf-ram  10r RECEIVE     1
  114692   1 proc/boot/devc-pty  10r RECEIVE     1
  114693   1 proc/boot/io-net    10r SIGWAITINFO
  114693   2 proc/boot/io-net    20r RECEIVE     5
  114693   3 proc/boot/io-net    10r RECEIVE     1
  114693   4 proc/boot/io-net    10r RECEIVE     1
  114693   5 proc/boot/io-net    10r RECEIVE     1
  114693   6 proc/boot/io-net    10r CONDVAR     97400914
  114693   8 proc/boot/io-net    10r RECEIVE     1
  114694   1 proc/boot/sh        10r SIGSUSPEND
  114695   1 proc/boot/pipe      10r RECEIVE     1
  114695   2 proc/boot/pipe      10r RECEIVE     1
  114695   3 proc/boot/pipe      10r RECEIVE     1
  114695   4 proc/boot/pipe      10r RECEIVE     1
  114698   1 proc/boot/inetd     10r SIGWAITINFO
  114699   1 proc/boot/WBSrvr    10r SIGWAITINFO
```

```
  118792   1 proc/boot/halMeDrv  10r RECEIVE     1
  118796   1 c/boot/sysmgr_g_ns  10r NANOSLEEP
  118796   2 c/boot/sysmgr_g_ns  11r INTR
  118798   1 proc/boot/rpcbind   10r SIGWAITINFO
  151561   1 roc/boot/ipcp_g_ns  10r RECEIVE     1
  151561   2 roc/boot/ipcp_g_ns  10r NANOSLEEP
  151561   3 roc/boot/ipcp_g_ns  10r NANOSLEEP
  151561   4 roc/boot/ipcp_g_ns  10r INTR
  155661   1 oc/boot/ha_hb_g_ns  10r JOIN        2
  155661   2 oc/boot/ha_hb_g_ns  60r CONDVAR     974007ac
  159759   1 c/boot/sdwrap_g_ns  10r CONDVAR     974008a8
  172048   1 boot/setClock_g_ns  10r CONDVAR     974007f4
  184337   1 oot/dumper_cp_g_ns  10r RECEIVE     1
  188434   1 /showProcInfo_g_ns  10r REPLY       1
  192531   1 t/loadBalance_g_ns  10r NANOSLEEP
  192531   2 t/loadBalance_g_ns  10r RECEIVE     1
  192531   3 t/loadBalance_g_ns  10r NANOSLEEP
  192531   4 t/loadBalance_g_ns  10r RECEIVE     4
  192531   5 t/loadBalance_g_ns  10r CONDVAR     97400788
  192531   6 t/loadBalance_g_ns  10s RECEIVE     8
  192531   7 t/loadBalance_g_ns  10s RECEIVE     12
  192531   8 t/loadBalance_g_ns  10s RECEIVE     16
  192531   9 t/loadBalance_g_ns  10s RECEIVE     20
  192531  10 t/loadBalance_g_ns  10s RECEIVE     24
  192531  11 t/loadBalance_g_ns  10r NANOSLEEP
  192531  12 t/loadBalance_g_ns  10r RECEIVE     36
  192531  13 t/loadBalance_g_ns  10r NANOSLEEP
  192531  14 t/loadBalance_g_ns  10r RECEIVE     39
  196628   1 t/inspectHttp_g_ns  10r NANOSLEEP
  196628   2 t/inspectHttp_g_ns  10r NANOSLEEP
  196628   3 t/inspectHttp_g_ns  10r CONDVAR     97400884
  200725   1 ot/appInspect_g_ns  10r SEM         a68e800
  200725   2 ot/appInspect_g_ns  10r NANOSLEEP
  204822   1 oc/boot/sslHs_g_ns  10r NANOSLEEP
  204822   2 oc/boot/sslHs_g_ns  10r CONDVAR     974008f0
  204822   3 oc/boot/sslHs_g_ns  10r CONDVAR     9740095c
  208919   1 /boot/me_dump_g_ns  10r SIGWAITINFO
  905240   1 proc/boot/sh        10r SIGSUSPEND
 4304921   1 proc/boot/sh        10r SIGSUSPEND
 5701658   1 proc/boot/sh        10r SIGSUSPEND
 6324251   1 proc/boot/sh        10r SIGSUSPEND
 7335964   1 proc/boot/sh        10r SIGSUSPEND
 7450653   1 proc/boot/p         10r NANOSLEEP
 7528478   1 proc/boot/sh        10r SIGSUSPEND
 7528479   1 sbin/pidin          10r REPLY       1
```

Notes

The following process information is for QNX processes running on the X-SCALE of the selected IXP.

```
================================
Per-Process Information
================================
  UID        PID       PPID        TIME COMMAND
    0          1          0    16:31:05
    0          2          1    00:22:58 devc-ser8250-ixp2400
    0          3          1    00:00:01 devf-ram
    0     114692          1    00:00:00 devc-pty
    0     114693          1    00:00:00 io-net
```

```
    0     114694          1    00:00:53 sh
    0     114695          1    00:00:00 pipe
    0     118792          1    00:00:00 halMeDrv
    0     151561          1    00:00:01 ipcp_g_ns
    0     114698          1    00:00:00 inetd
    0     114699          1    00:00:00 WBSrvr
    0     118796          1    00:00:00 sysmgr_g_ns
    0     155661          1    00:00:00 ha_hb_g_ns
    0     118798          1    00:00:00 rpcbind
    0     159759          1    00:00:00 sdwrap_g_ns
    0     172048          1    00:00:00 setClock_g_ns
    0     184337          1    00:00:00 dumper_cp_g_ns
    0     188434          1    00:00:00 showProcInfo_g_ns
    0     192531          1    00:00:06 loadBalance_g_ns
    0     196628          1    00:00:00 inspectHttp_g_ns
    0     200725          1    00:00:00 appInspect_g_ns
    0     204822          1    00:00:03 sslHs_g_ns
    0     208919          1    00:00:00 me_dump_g_ns
    0     905240     114694    00:04:40 sh
    0    4304921     905240    00:01:44 sh
    0    5701658    4304921    00:00:56 sh
    0    6324251    5701658    00:01:26 sh
    0    7335964    6324251    00:00:08 sh
    0    7450653    7335964    00:00:00 p
    0    7553054     188434    00:00:00 sh
    0    7553055    7553054    00:00:00 /sbin/ps
```

The following MicroEngine utilization is *derived* by examining the idle statistics for the queues from which the particular ME reads data. If the given queue is not being read, then the utilization for the particular ME is set to *100*. This is a flag value rather an actual indication that the CPU is running at 100%. For instance, a deadlock condition could cause a 100 utilization when the ME is actually not processing any data at all.

```
ME Utilization Statistics
--------------
RECEIVE:                                        0
FASTPATH:                                       0
SLOWTX:                                         0
REASSEMBLY:                                     0
TCP_RX:                                         0
HTTP:                                           0
IH_RX                                           0
SSL_ME:                                         0
CM_CLOSE:                                       0
X_TO_ME:                                        0
FIXUP:                                          0
OCM:                                            0
TCP_TX:                                         0
ICM:                                            0
```

show np 1 interface icmlookup

Displays the Inbound Connection Manager (ICM)/Outbound Connection Manager (OCM) interface table from the Control Plane (CP), number 0, or the specified NP. This is a listing of configured VLANs from the perspective of the ICM process of the ACE. This table is used by ICM to process and/or forward packets as needed.

Sample Output

```
switch/Admin# show np 1 interface icmlookup
ICM Lookup Table:
L2 ACL: BPDU IPV6 MPLS all
Flags: Status FT-status FT-vlan Bridged RPF Stick-src-mac normalization icmp-guard switch-mode

ifid vlan ctx ftg bg   oif bvid iacl oacl rt IPAddress        MAC          l2acl Flags
---- ---- --- --- ---- --- ---- ---- ---- -- ---------        ------------ ----- -----
1    1    0   1   1    0   0    1    0    0  127.1.0.1         0019aaccbfd5 0000  110000000
2    5    0   1   2    0   0    2    0    0  10.86.215.35      0019aaccbfd5 0000  110000000
6    10   0   1   6    0   0    5    6    0  192.168.1.129     0019aaccbfd5 0000  110000000
7    20   0   1   7    0   0    7    0    0  192.168.2.129     0019aaccbfd5 0000  110000000
8    40   0   1   8    0   0    8    0    0  209.165.201.3     0019aaccbfd5 0000  111000000

Vlan       Thresholds        |            Processed
       Configured   Current  |    SYNs     ACKsSucc   ACKsFail
----   ---------- ---------- | ---------- ---------- ----------
1      0x0        0x0          0x0        0x0        0x0
5      0x0        0x0          0x0        0x0        0x0
10     0x0        0x0          0x0        0x0        0x0
20     0x0        0x0          0x0        0x0        0x0
40     0x0        0x0          0x0        0x0        0x0
```

Notes

Field	Description
ifid	Internal Interface Identifier or "ifIndex" for the configured VLAN. The ifIndex is a unique internal identified of each VLAN.
vlan	The configured VLAN number.
ctx	The context identifier (ID). This can be matched up from the "show context" ouptut.
ftg	If redundancy is configured, this is the FT group number from the configuration. This is user-assigned.
bg	Bridge Group Indentifier. The default is the configured VLAN number but can be modified via the configuration.
oif	If Fault Tolerant (FT) is configured this is the Internal Interface Identifier or "ifIndex" for the peer VLAN.
bvid	The Bridge-Group Virtual Interface Indentification Number (BVID) from the configuration.
iacl	The internal INPUT identifier for the Access Control List (ACL) to ICM from this VLAN.
oacl	The internal OUTPUT identifier for the Access Control List (ACL) to ICM from this VLAN.
rt	The Route Identifier (ID) for ICM.
IPAddress	The configured IP address on this VLAN.
MAC	The assigned MAC address for this VLAN.
l2acl	The four bits for the Layer 2 (L2) ACL, which are, in order, BPDU (Spanning Tree), IPV6 (Not Supported), MPLS (MultiProtocol Label Switching) or ALL (Not Supported).
Flags	The ICM Interface Flags, which are 9 bits in length and in the following order: • Status ? 1-enabled, 0-disabled

	• FT-status ? 1-enabled, 0-disabled • FT-vlan ? 1-configured, 0-not configured • Bridged ? 1-yes, 0-no • RPF ? 1-enabled, 0-disabled; RFP is Reverse Path Forwarding for multicasts. • Stick-src-mac ? 1-configured 0-not configured) • normalization ? 1-configured, 0-not configured • icmp-guard ? 1-configured, 0-not configured • switch-mode ? 1-configured, 0-not configured
Configured Threshold	The configured embryonic connection threshold above which the ACE applies SYN-cookie Denial of Service (DoS) protection. Range 1 to 65535. This and the following counters are related to the configuration of the SYN Cookie feature on ACE.
Current Threshold	The calculated threshold observed on this VLAN.
Processed SYNs	Number of SYNs handled for SYN Cookie.
Processed ACKsSucc	Number of successful SYN Cookie Connections.
Processed ACKsFail	Number of failed SYN Cookie Connections.

show np 1 interface iflookup

This command displays the fastpath interface lookup table from the CP(0) or the specified NP.

Sample Output

```
ACE30001/Admin# show np 0 interface iflookup
Hostid: 10
Shared vlan macs currently in use (offset from 10240): 0-7
Vlan-vmac indexes currently in use: 0-4
Flags:  Valid shared bridged ftstatus ssl-test normalization icmp-guard switch-mode

Vlan   ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
----   ---- ------- ---- ------- ----- ----- --- ------ --    -----
1      1    1       0    1       1     1     0   2      0     1101000
23     3    3       0    23      2     1     0   2      0     1101110
101    4    4       0    101     3     1     0   2      0     1101000
102    2    2       0    102     4     1     0   2      0     1101000
103    16   16      1    103     0     4     0   2      0     1001000
104    5    5       0    104     0     1     0   2      0     1001000
105    15   15      2    105     0     3     0   2      0     1001000
128    9    9       3    128     0     2     0   2      0     1000000
192    17   17      1    192     0     4     0   2      0     1001000

Vlan       Thresholds          |            Processed
       Configured   Current   |    SYNs     ACKsSucc   ACKsFail
----   ---------- ---------- | ---------- ---------- ----------
1      0x0        0x0          0x0        0xd0000000 0x10000
23     0x62910000 0xd0000000   0x660000   0x10004    0x0
101    0x20000    0xc0a802fa   0x1d70d1   0x62910000 0xd0180000
102    0x10001    0x0          0x10000    0x7f010001 0x1d70d1
103    0xc0000003 0x170000     0x20006    0x3        0xa0000
104    0x170000   0x10002      0x0        0x30000    0xa56d7b2
105    0x3        0x90000      0xabc5036  0x1d70d1   0x62910000
128    0xc0a804b2 0x1d70d1     0x62910000 0xd0000001 0x10000
```

```
192    0xa56d7f3  0x1243dc     0x93030000 0xc0000003 0x650000

MACidx ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
------ ---- ------- ---- ------- ----- ----- --- ------ --    -----
14336  6    6       1    1       1     4     0   2      0     1101000
14337  7    7       2    1       1     3     0   2      0     1101000
14338  8    8       3    1       1     2     0   2      0     1100000
14339  10   10      3    23      2     2     0   2      0     1100000
14340  11   11      3    101     3     2     0   2      0     1100000
14342  13   13      2    102     4     3     0   2      0     1101000
14343  14   14      2    101     3     3     0   2      0     1101000

VVind  ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
------ ---- ------- ---- ------- ----- ----- --- ------ --    -----
1      1    1       0    1       1     1     0   2      0     1101000
4      2    2       0    102     4     1     0   2      0     1101000
2      3    3       0    23      2     1     0   2      0     1101110
3      4    4       0    101     3     1     0   2      0     1101000
1      6    6       1    1       1     4     0   2      0     1101000
1      7    7       2    1       1     3     0   2      0     1101000
4      13   13      2    102     4     3     0   2      0     1101000
3      14   14      2    101     3     3     0   2      0     1101000


ACE30001/Admin# show np 1 interface iflookup
First burnt-in MAC: 00:1d:70:d1:62:91
Last  burnt-in MAC: 00:1d:70:d1:62:97
No of burnt-in MACs: 7
Hostid: 10
Shared vlan macs currently in use (offset from 10240): 0-7
Vlan-vmac indexes currently in use: 0-4
Flags:  Valid shared bridged ftstatus ssl-test normalization icmp-guard switch-mode

Vlan   ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
----   ---- ------- ---- ------- ----- ----- --- ------ --    -----
1      1    1       0    1       1     1     0   2      0     1101000
23     3    3       0    23      2     1     0   2      0     1101110
101    4    4       0    101     3     1     0   2      0     1101000
102    2    2       0    102     4     1     0   2      0     1101000
103    16   16      1    103     0     4     0   2      0     1001000
104    5    5       0    104     0     1     0   2      0     1001000
105    15   15      2    105     0     3     0   2      0     1001000
128    9    9       3    128     0     2     0   2      0     1000000
192    17   17      1    192     0     4     0   2      0     1001000

Vlan       Thresholds         |            Processed
       Configured   Current   |    SYNs     ACKsSucc   ACKsFail
----   ---------- ---------- | ---------- ---------- ----------
1      0x0        0x0         0x0        0x0        0x0
23     0x0        0x0         0x0        0x0        0x0
101    0x0        0x0         0x0        0x0        0x0
102    0x0        0x0         0x0        0x0        0x0
103    0x0        0x0         0x0        0x0        0x0
104    0x0        0x0         0x0        0x0        0x0
105    0x0        0x0         0x0        0x0        0x0
128    0x0        0x0         0x0        0x0        0x0
192    0x0        0x0         0x0        0x0        0x0

MACidx ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
------ ---- ------- ---- ------- ----- ----- --- ------ --    -----
14336  6    6       1    1       1     4     0   2      0     1101000
14337  7    7       2    1       1     3     0   2      0     1101000
14338  8    8       3    1       1     2     0   2      0     1100000
14339  10   10      3    23      2     2     0   2      0     1100000
```

```
14340  11   11      3    101     3     2     0   2      0     1100000
14342  13   13      2    102     4     3     0   2      0     1101000
14343  14   14      2    101     3     3     0   2      0     1101000

VVind  ifid matchid ctxt primary vvind ftgrp ttl optact df    Flags
------ ---- ------- ---- ------- ----- ----- --- ------ --    -----
1      1    1       0    1       1     1     0   2      0     1101000
4      2    2       0    102     4     1     0   2      0     1101000
2      3    3       0    23      2     1     0   2      0     1101110
3      4    4       0    101     3     1     0   2      0     1101000
1      6    6       1    1       1     4     0   2      0     1101000
1      7    7       2    1       1     3     0   2      0     1101000
4      13   13      2    102     4     3     0   2      0     1101000
3      14   14      2    101     3     3     0   2      0     1101000
```

Notes

Field	Description
Vlan	The configured VLAN identifier (ID).
ifid	Internal interface identifier or "ifIndex" of the configured VLAN. The ifIndex is a unique, internal identifier for each VLAN.
matchid	For this command output, this value will always be the same as the "ifid".
ctxt	The context identifier (id). This can be matched up from the "show context" output.
primary	The configured VLAN ID.
vvind	The virtual VLAN ID if the VLAN is allocated to other configured contexts.
ftgrp	If Fault Tolerant (FT) is configured, this is the FT group number from the configuration. This is a user-assigned value.
ttl	The default IPV4 Time To Live (TTL) for packets generated by the ACE on this interface/VLAN.
optact	Whether the interface supports IPV4 options.
df	Whether the interface always clears the IPV4 Don't Fragment (DF) bit in IPV4 packets.
Flags	A 7-bit field with the flags in the following order: • Valid ? 1-valid, 0-not valid • shared ? 1-yes, 0-no • bridged ? 1-yes, 0-no • ftstatus ? 1-enabled, 0-disabled • ssl-test ? 1-enabled, 0-disabled; DEBUG only • normalization ? (1-configured 0-not configured) • icmp-guard ? 1-configured, 0-not configured • switch-mode ? 1-configured, 0-not configured
Vlan Configured/Current Thresholds	The SYN Cookie Feature for Denial of Service (DOS) protection. See the description of the "show np 1 interface icmlookup" output for more details.

The final two groups of output display the same interface information by the MAC index (MACidx) and the Virtual VLAN ID (VVind). The reset of the fields are the same.