Contents

show acl-merge merged-list

Displays the acl-merge list per-context for a specified VLAN. The ACL-merge list is a single ACL (access control list) that the CP compiles from multiple security ACL entries and policies present in the configuration. The information displayed by this command represents the actions that the ACE will perform on a flow based on this acl-merged list.

Sample Output

```
ACE30001/Admin# show acl-merge merged-list vlan 23 in

--------------
Context ID: 0
--------------
All ACEs in merged list 2 Total:7 Non-redundant:7

Priority:16000, Lineno:2, ACE-id:147 Action:PERMIT, Path-id:0x81/0x0/0x81:6/0[6/
0][6/0]
Pmap:0x4, Log:FALSE/FALSE[FALSE][FALSE], Interval:0/0[0][0]
Hash1:0x0 Hash2:0x0
Generated:FALSE, need-to-add-in-comp:NO_ACT_NEEDED, redundant:FALSE
Parent:: feature:TO CP ace-lineno:2 ACL priority:16779265[G:0,P:1,C:8,ACL:1]
Feature:TO CP Policy:1[1][1] sec-level:0x0 Intratype:TERMINATE
Intertype:TERMINATE
IP address SRC:0.0.0.0/0.0.0.0 DST:10.86.215.178/255.255.255.255
Ports SRC:RANGE 0 65535 DST:RANGE 22 22
Protocol:6
Hit Count:0 Active:TRUE Timerange:0

Priority:32000, Lineno:3, ACE-id:148 Action:PERMIT, Path-id:0x81/0x0/0x81:6/0[6/
0][6/0]
Pmap:0x4, Log:FALSE/FALSE[FALSE][FALSE], Interval:0/0[0][0]
Hash1:0x0 Hash2:0x0
Generated:FALSE, need-to-add-in-comp:NO_ACT_NEEDED, redundant:FALSE
Parent:: feature:TO CP ace-lineno:3 ACL priority:16779265[G:0,P:1,C:8,ACL:1]
Feature:TO CP Policy:1[1][1] sec-level:0x0 Intratype:TERMINATE
Intertype:TERMINATE
IP address SRC:0.0.0.0/0.0.0.0 DST:10.86.215.178/255.255.255.255
Ports SRC:RANGE 0 65535 DST:RANGE 23 23
Protocol:6
Hit Count:0 Active:TRUE Timerange:0
```

show arp statistics

Displays statistics related to Address Resolution Protocol activities in ACE. The values shown represent the total for all services in a given context.

Sample Output

```
ace3/Admin# show arp statistics

Context:Admin
```

```
RX Packets              : 1275          RX Errors               : 0
TX Packets              : 140           TX Errors               : 0
Bridged Packets         : 0             Bridged Errors          : 0
Requests Recvd          : 2             Requests Sent           : 134
Response Recvd          : 13            Response Sent           : 2
Packets dropped         : 1260          Inspect failed          : 0
Collisions Detected     : 0             Gratuitous ARP sent     : 4
Hosts learned           : 2             Smac-validation failed  : 0
Resolution Requests     : 0             Encap-miss msg          : 0
Pings attempted for Encap-miss msg              : 0
Pings quenched for Encap-miss msg               : 0
Pings rejected for Encap-miss msg               : 0
Pinged Encap-miss responded to                  : 0
Replication Counters:
--------------------
Msg Received            : 0
Hosts Replicated        : 0
Replication Failed      : 0
Replication Ignored     : 0
```

Notes

Field	Description
RX Packets	Packets received
RX Errors	Packets received with errors
TX Packets	Packets transmitted
TX Errors	Packet transmission errors
Bridged Packets	Packets bridged
Bridged Errors	Bridged packets that had errors
Requests Recvd	ARP requests received
Requests Sent	ARP requests sent
Response Recvd	ARP responses received
Response Sent	ARP responses sent
Packets dropped	Packets were dropped. Typically, the "Packets dropped" value is slightly less than "RX packets" value during normal operation. Reasons for packet dropping include the following: • Invalid source IP/MAC address • The flood ARP handling option is not enabled and the interface is not bridging. (This counter will be incremented along with the "Smac-validation failed" counter.) If debug mode is enabled, these error messages are produced: ♦ debug arp err ? "ARP DAI checks failed. Dropping pkt!" ♦ debug arp err ? "Dropping pkt with smac-validation failure..." • Packet received on the standby and not for the standby IP. If debug mode is enabled, this error message is produced: debug arp info - "::Cannot bridge or process ARP pkt ip:xxx". • No peer IP address is configured. If debug mode is enabled, this error message is produced: debug arp err ? ":: Failed to get peer-ip address. ifid:xxx" • A MAC collision occurred for a static mac-entry. • The entry is in the ARP cache but a failure occurred in getting the MAC address from ARP entry. If debug mode is enabled, this error message is produced: debug arp err ? "Failed to Get mac addess from ARP entry!".

	• The address is "owned" by the ACE because it is an interface address of the active/standby or any other local address (alias, vip, nat) on the active. • ARP cache entry creation failed. • If the entry is not in the ARP cache or if it is a local entry and is not bridged. • If bridging and if sourced from or destined to the standby ACE. If debug mode is enabled, this error message is produced: debug arp info ? "Dropping Packet source/dest ip xxxx is of standby module". • Drop ARP response packet if the destination entry is found on the same interface as the source, or is a unicast flood packet. If debug mode is enabled, this error message is produced: debug arp err ? ":: Dropping Response Packet.. unicast flood". • The entity_type is unknown (this is an internal error). • Failed to get the source ARP entry from the ARP cache (for responses). If debug mode is enabled, this error message is produced: debug arp err ? "Unable to locate MAC addr for ip: xxx intf: xxx". • Could not create response for ARP request. If debug mode is enabled, this error message is produced: debug arp err ? "Could not Create Response for ARP request for". • Received rarp in ARP handler. If debug mode is enabled, this error message is produced: debug arp info ? "received rarp in arp handler". • Received unknown ARP type (unknown opcode). (The only valid values are REQUEST (0x1) and REPLY(0x2).) If debug mode is enabled, this error message is produced: debug arp err ? "received unknown arp type xx".
Inspect failed	Packets failed inspection
Collisions Detected	Collisions detected
Gratuitous ARP sent	Gratuitous ARPs sent
Hosts learned	Number of host IP addresses that were learned
Smac-validation failed	Number of times ARP requests were received with same MAC address
Resolution Requests	This counter does not track the received ARP requests on any VLAN interface but keeps track of an internal ACE event, it counts the MTS requests "MTS_OPC_ITASCA_ARP_RESOLUTION", which is the MTS request to resolve ARP for the PEER IP. It is triggered by a heartbeat message. And a Resolution Request message gets sent which updates the arp cache if needed.
Encap-miss msg	Whenever the DP (ICM or OCM) encounters an encaps ID of 0 for packets, it makes an IPCP call to the internal ARP Manager to resolve the encaps ID. This counter indicates the number on the encap miss message sent by DP to CP for encap resolution. After receiving Encap-miss message, if the IP address is directly connected to ACE, the ARP Manager sends an ARP request to get the mac resolved for the IP else if the IP is not directly reachable, ACE sends a ping message to the IP address; all the stats below are related to when the ARP Manager sends the ping to resolve the MAC of the next hop.
Pings attempted for Encap-miss msg	Number of times that the ACE recognizes that a ping attempt needs to occur when an Encap miss due to destination packet IP address not on an existing bridge-group subnet occurs.
Pings quenched for Encap-miss msg	Number of times that the ACE suppresses an effort to ping for the same destination packet IP address if the Encap miss for that address occurs repeatedly and too fast.

Pings rejected for Encap-miss msg	Number of times that the ACE rejects ping attempts for destination IP addresses when the Encap misses for that address are too many to handle. Similar to the quenched pings, these misses are unique.
Pinged Encap-miss responded to	Number of actual pings sent for a missed IP address. The number of this counter should match the number of pings that were attempted for the Encap-miss message counter.
Replication Counters	These counters are related to the number of messages exchanged between active and standby. Standby gets sync messages for hosts which are learnt by the active.

show buffer event-history

This command is primarily intended for internal use. It displays a historic log of the most recent messages generated by the diagnostic buffer event manager. It is used in conjunction with the diagnostic command **debug buffer**.

Note that the buffers referenced in the command are zero-copy buffers shared between the ACE user-space and the ACE kernel drivers.

The **debug buffer** command has the following usage:

```
switch/Admin# debug buffer ?
  all      Debug CP buffer all
  error    Debug CP buffer errors
  info     Debug CP buffer info
  warning  Debug CP buffer warnings
```

Sample Output

```
switch/Admin# show buffer event-history
1) Event:E_DEBUG, length:72, at 532056 usecs after Sat Jan  1 00:00:25 2000
    [102] headers=0xd2385000, ctrl_blocks=0xd2825260, data_blocks=0xd54122e0

2) Event:E_DEBUG, length:50, at 532034 usecs after Sat Jan  1 00:00:25 2000
    [102] total blocks=151512 (ctrl=75756, data=75756)
```

Notes

The output shows:

- The hexadecimal numbers printed are ACE kernel virtual addresses indicating where the buffers are located.
- The two buffer pool virtual addresses for the control (ctrl) and data buffer pools.

show buffer stats

This command shows detailed counters for various buffer manager event occurrences. Specifically, it shows statistics for the control plane's buffer, with stats for DEFAULT_CONTROL pool, DEFAULT_DATA pool (which are automatically created at initialization) and total count.

You should provide the output of this command, along with that of **show buffer usage**, to Cisco TAC in the event of buffer manager errors, for instance, as indicated by the error message: "No memory from buffer manager. Cannot send packet."

Sample Output

```
Control Plane Buffer Statistics
-------------------------------
Pool Name: DefaultCtrl , Priority: High
Total Buffers   : 75756        In Use       : 32768
Total Allocated : 53101        Hi Watermark : 75756
Total Freed     : 20333        Lo Watermark : 42986
Alloc Failures  : 0

Pool Name: DefaultData , Priority: Normal
Total Buffers   : 75756        In Use       : 32768
Total Allocated : 161580       Hi Watermark : 75756
Total Freed     : 128812       Lo Watermark : 42979
Alloc Failures  : 0

Totals
Buffers : 151512       Allocated : 214681
In Use  : 65536        Freed     : 149145
```

Notes

Field	Description
Total Buffers	Maximum number of buffers in a given pool.
Total Allocated	Total number of buffer allocated up until now, where some of them may be freed now. Actually "Total Allocated = buffer in use + total freed".
Total Freed	Total number of buffer freed till now. Actually "Total Freed = Total Allocated - buffer in use".
In use	Number of buffers currently being used.
Alloc Failures	Number of buffer allocations that failed.
Hi Watermark	This is the maximum value of available buffers ever. That is, the "max of (Total Buffers - In use) all time".
Lo watermark	This is the lowest value of available buffers ever. That is, the "min of (Total Buffers - In use) all time".

show buffer usage

This command displays the number of buffers currently being held (allocated but not freed) by each buffer module. The "Multiple Frees" column shows an estimate of the number of times a particular buffer module has freed the same buffer more than once (this indicates a software error condition).

The **show buffer usage** command displays the per-owner usage array. This is useful for identifying error conditions in which the module is not freeing buffers or freeing the same buffer multiple times.

You should provide the output of this command, along with that of **show buffer stats**, to Cisco TAC in the event of buffer manager errors, for instance, as indicated by the error message: "No memory from buffer manager. Cannot send packet."

Sample Output

```
Module              Current Usage    Multiple Frees
------              -------------    --------------
```

```
Unknown             0                0
Test Utilities      0                0
Pkt Fifo Driver     65536            0
VNet Driver         0                0
IPCP                0                0
Encap/Decap         0                0
Arp Manager         0                0
Health Monitor      0                0
ICMP Manager        0                0
BPDU Handler        0                0
Session Filter      0                0
IF Manager          0                0
```

Notes

Resource	Maximum Value
Module	The owner who is using the buffer space.
Current usage	Number of buffers being used by a given owner currently.
Multiple frees	Number of "Multiple free" events by a given owner up until now.
Unknown	Unknown/invalid owner
Test Utilities	A module to test buffer allocation/free from kernel module context; it uses the buffer for testing the kernel programs.
Pkt Fifo Driver	FIFO driver module. As shown in the sample output, this is typically a large value.
VNet Driver	Linux Pseudo-Driver module.
IPCP	IPCP driver module.
Arp Manager	ARP Manager events; usually incremented by ARP requests and responses.
BPDU Handler	BPDU fixup/forwarding handler module.
Session Filter	IXP Session Filter module.

show cde all

Displays the values of all Classification Distribution Engine (CDE) registers. The CDE is a component within the ACE module that acts as a central point of contact between all the main components in the module. It is a field programmable gate array (FPGA) that can be thought of as a smart switch within the ACE; it decides where an incoming packet should be sent among the various components on the ACE.

Several show commands provide information on the status of the CDE. A few notes on these commands:

- They are module-specific
- They can only be performed in the admin context
- Except for the **show cde health** command, these commands are primarily used for internal development purposes and not relevant to general troubleshooting. However, they are listed here for completeness.

Sample Output

```
switch/Admin# show cde all
cde1 reg 0x   0                       CD_CP_RST val 0x1
```

```
cde1 reg 0x   1                      CD_CP_RID val 0x403
cde1 reg 0x   2                  CD_CP_ERR_INT val 0x0
cde1 reg 0x   3              CD_CP_ERR_INT_MSK val 0x3
cde1 reg 0x   4                      CD_CP_CFG val 0xc
cde1 reg 0x   7                CD_CP_ERR_STATE val 0x0
cde1 reg 0x  80                      CD_DH_CFG val 0x0
cde1 reg 0x  81              CD_DH_CP_MDT_ADDR val 0xf
cde1 reg 0x  82           CD_DH_CP_MDT_DATA__3 val 0x1
cde1 reg 0x  83           CD_DH_CP_MDT_DATA__2 val 0x0
cde1 reg 0x  84           CD_DH_CP_MDT_DATA__1 val 0x0
cde1 reg 0x  85           CD_DH_CP_MDT_DATA__0 val 0x0
cde1 reg 0x  86               CD_DH_CP_VT_ADDR val 0x1e
cde1 reg 0x  87               CD_DH_CP_VT_DATA val 0x3
cde1 reg 0x  88              CD_DH_CP_RBH_ADDR val 0x1f
cde1 reg 0x  89              CD_DH_CP_RBH_DATA val 0x2
cde1 reg 0x  10                      CD_HR_CFG val 0x380
cde1 reg 0x  11            CD_HR_THRESHOLD_CFG val 0x248
cde1 reg 0x  12              CD_HR_DST_ENB_CFG val 0xd6
cde1 reg 0x  13                CD_HR_ROUTE_CFG val 0x141
cde1 reg 0x  14                 CD_HR_IRH_CFG0 val 0x0
cde1 reg 0x  15                 CD_HR_IRH_CFG1 val 0x3
cde1 reg 0x  16        CD_HR_IRH_ADDR_UPPER_CFG val 0x0
cde1 reg 0x  17        CD_HR_IRH_ADDR_LOWER_CFG val 0x0
cde1 reg 0x  1d                    CD_HR_INT0 val 0x0
cde1 reg 0x  1e                CD_HR_INT0_MSK val 0xffff
cde1 reg 0x  1f                    CD_HR_INT1 val 0x100
cde1 reg 0x  20                CD_HR_INT1_MSK val 0x2ff
cde1 reg 0x  25                  CD_HR_STATUS val 0x0
cde1 reg 0x  c0                CD_HT_DHDR_CFG0 val 0xf00
cde1 reg 0x  c1                CD_HT_DHDR_CFG1 val 0x0
cde1 reg 0x  c2                CD_HT_DHDR_CFG2 val 0x0
cde1 reg 0x  c3                CD_HT_DHDR_CFG3 val 0x0
cde1 reg 0x  c4           CD_HT_DHDR_SRC_CFG__1 val 0x0
cde1 reg 0x  c5           CD_HT_DHDR_SRC_CFG__0 val 0x80
cde1 reg 0x  c6           CD_HT_DHDR_DST_CFG__1 val 0x0
cde1 reg 0x  c7           CD_HT_DHDR_DST_CFG__0 val 0x0
cde1 reg 0x  ca                    CD_HT_INT val 0x0
cde1 reg 0x  cb                CD_HT_INT_MSK val 0x7ff
cde1 reg 0x  cd             CD_HT_IMPH_DBG_CFG val 0x0
cde1 reg 0x  ce                  CD_HT_STATUS  val 0xf800
cde1 reg 0x 180               CD_SI0_SRC_STATUS val 0x1
cde1 reg 0x 181               CD_SI0_SRC_CONFIG val 0x448
cde1 reg 0x 182     CD_SI0_SRC_AF_THRESH_ASSERT val 0x10
cde1 reg 0x 183     CD_SI0_SRC_AF_THRESH_NEGATE val 0x20
cde1 reg 0x 184              CD_SI0_SRC_CAL_LEN val 0x13
cde1 reg 0x 185             CD_SI0_SRC_CAL_ADDR val 0x13
cde1 reg 0x 186             CD_SI0_SRC_CAL_DATA val 0x8
cde1 reg 0x 187               CD_SI0_SNK_STATUS val 0x5
cde1 reg 0x 188               CD_SI0_SNK_CONFIG val 0x8
cde1 reg 0x 189     CD_SI0_SNK_AF_THRESH_ASSERT val 0x10
cde1 reg 0x 18a     CD_SI0_SNK_AF_THRESH_NEGATE val 0x20
cde1 reg 0x 18b              CD_SI0_SNK_CAL_LEN val 0x13
cde1 reg 0x 18c             CD_SI0_SNK_CAL_ADDR val 0x13
cde1 reg 0x 18d             CD_SI0_SNK_CAL_DATA val 0x8
cde1 reg 0x 18e          CD_SI0_SNK_MISC_CONFIG val 0x11
cde1 reg 0x 197                  CD_SI0_SRC_INT val 0x0
cde1 reg 0x 198               CD_SI0_SRC_INT_MSK val 0xf
cde1 reg 0x 199               CD_SI0_SNK_INT__1 val 0x0
cde1 reg 0x 19a               CD_SI0_SNK_INT__0 val 0x0
cde1 reg 0x 19b            CD_SI0_SNK_INT_MSK__1 val 0xf
cde1 reg 0x 19c            CD_SI0_SNK_INT_MSK__0 val 0xffff
cde1 reg 0x 100            CD_XS1_BI_DIS_CRC_CHK val 0x0
cde1 reg 0x 101                 CD_XS1_XX_GP_CFG val 0x0
cde1 reg 0x 102           CD_XS1_GLOBAL_TRAP_CFG val 0x427
```

```
cde1 reg 0x 103                    CD_XS1_DB_CFG val 0xf
cde1 reg 0x 104             CD_XS1_DI_CRC_ERR_INT val 0x0
cde1 reg 0x 105          CD_XS1_DI_CRC_ERR_INT_MSK val 0x3
cde1 reg 0x 106               CD_XS1_XX_ENQ_INT__3 val 0x0
cde1 reg 0x 107               CD_XS1_XX_ENQ_INT__2 val 0x0
cde1 reg 0x 108               CD_XS1_XX_ENQ_INT__1 val 0x0
cde1 reg 0x 109               CD_XS1_XX_ENQ_INT__0 val 0x0
cde1 reg 0x 10a           CD_XS1_XX_ENQ_INT_MSK__3 val 0xff
cde1 reg 0x 10b           CD_XS1_XX_ENQ_INT_MSK__2 val 0xffff
cde1 reg 0x 10c           CD_XS1_XX_ENQ_INT_MSK__1 val 0xffff
cde1 reg 0x 10d           CD_XS1_XX_ENQ_INT_MSK__0 val 0xffff
cde1 reg 0x 10e               CD_XS1_XX_VOQ_INT__2 val 0x0
cde1 reg 0x 10f               CD_XS1_XX_VOQ_INT__1 val 0x0
cde1 reg 0x 110               CD_XS1_XX_VOQ_INT__0 val 0x0
cde1 reg 0x 111           CD_XS1_XX_VOQ_INT_MSK__2 val 0x3
cde1 reg 0x 112           CD_XS1_XX_VOQ_INT_MSK__1 val 0xffff
cde1 reg 0x 113           CD_XS1_XX_VOQ_INT_MSK__0 val 0xffff
cde1 reg 0x 114          CD_XS1_XX_VOQ_PERR_INT__1 val 0x0
cde1 reg 0x 115          CD_XS1_XX_VOQ_PERR_INT__0 val 0x0
cde1 reg 0x 116    CD_XS1_XX_VOQ_PERR_INT_MSK__1 val 0x1
cde1 reg 0x 117    CD_XS1_XX_VOQ_PERR_INT_MSK__0 val 0xffff
cde1 reg 0x 118             CD_XS1_XX_DI_TRAP_INT val 0x0
cde1 reg 0x 119          CD_XS1_XX_DI_TRAP_INT_MSK val 0x3fff
cde1 reg 0x 11a                    CD_XS1_CC_INT val 0x0
cde1 reg 0x 11b                CD_XS1_CC_INT_MSK val 0x3
cde1 reg 0x 11c             CD_XS1_BI_CRC_ERR_INT val 0x0
cde1 reg 0x 11d          CD_XS1_BI_CRC_ERR_INT_MSK val 0x1
cde1 reg 0x 11e                  CD_XS1_XX_GP_STA val 0x1c
cde1 reg 0x 11f               CD_XS1_XX_RPULL_STA val 0x33
cde1 reg 0x 120                 CD_XS1_HYP_FC_STA val 0x0
cde1 reg 0x 121                 CD_XS1_IX0_FC_STA val 0x0
cde1 reg 0x 122                  CD_XS1_CC_FC_STA val 0x0
cde1 reg 0x 123                 CD_XS1_BCM_FC_STA val 0x0
cde1 reg 0x 124                  CD_XS1_DC_FC_STA val 0x0
cde1 reg 0x 125              CD_XS1_XX_VOQ_STA__2 val 0x1
cde1 reg 0x 126              CD_XS1_XX_VOQ_STA__1 val 0x5000
cde1 reg 0x 127              CD_XS1_XX_VOQ_STA__0 val 0x3fff
cde1 reg 0x 151           CD_XS1_CD_XS1_FC_INT__1 val 0x0
cde1 reg 0x 152           CD_XS1_CD_XS1_FC_INT__0 val 0x0
cde1 reg 0x 153      CD_XS1_CD_XS1_FC_INT_MSK__1 val 0x0
cde1 reg 0x 154      CD_XS1_CD_XS1_FC_INT_MSK__0 val 0x0
cde2 reg 0x   0                       CD_CP_RST val 0x1
cde2 reg 0x   1                       CD_CP_RID val 0x402
cde2 reg 0x   2                   CD_CP_ERR_INT val 0x0
cde2 reg 0x   3                CD_CP_ERR_INT_MSK val 0x3
cde2 reg 0x   4                       CD_CP_CFG val 0x0
cde2 reg 0x   5                   CD_CP_PM_ADDR val 0x0
cde2 reg 0x   6                   CD_CP_PM_DATA val 0x0
cde2 reg 0x   7                 CD_CP_ERR_STATE val 0x0
cde2 reg 0x 180                CD_SI0_SRC_STATUS val 0x1
cde2 reg 0x 181                CD_SI0_SRC_CONFIG val 0x498
cde2 reg 0x 182      CD_SI0_SRC_AF_THRESH_ASSERT val 0x1d0
cde2 reg 0x 183      CD_SI0_SRC_AF_THRESH_NEGATE val 0x1e0
cde2 reg 0x 184               CD_SI0_SRC_CAL_LEN val 0xf
cde2 reg 0x 185              CD_SI0_SRC_CAL_ADDR val 0xf
cde2 reg 0x 186              CD_SI0_SRC_CAL_DATA val 0x8
cde2 reg 0x 187                CD_SI0_SNK_STATUS val 0x5
cde2 reg 0x 188                CD_SI0_SNK_CONFIG val 0x38
cde2 reg 0x 189      CD_SI0_SNK_AF_THRESH_ASSERT val 0x20
cde2 reg 0x 18a      CD_SI0_SNK_AF_THRESH_NEGATE val 0x30
cde2 reg 0x 18b               CD_SI0_SNK_CAL_LEN val 0xf
cde2 reg 0x 18c              CD_SI0_SNK_CAL_ADDR val 0xf
cde2 reg 0x 18d              CD_SI0_SNK_CAL_DATA val 0x8
cde2 reg 0x 18e           CD_SI0_SNK_MISC_CONFIG val 0x11
```

```
cde2 reg 0x 197                  CD_SI0_SRC_INT val 0x0
cde2 reg 0x 198              CD_SI0_SRC_INT_MSK val 0xf
cde2 reg 0x 199               CD_SI0_SNK_INT__1 val 0x0
cde2 reg 0x 19a               CD_SI0_SNK_INT__0 val 0x0
cde2 reg 0x 19b           CD_SI0_SNK_INT_MSK__1 val 0xf
cde2 reg 0x 19c           CD_SI0_SNK_INT_MSK__0 val 0xffff
cde2 reg 0x 1c0               CD_SI1_SRC_STATUS val 0x1
cde2 reg 0x 1c1               CD_SI1_SRC_CONFIG val 0x448
cde2 reg 0x 1c2      CD_SI1_SRC_AF_THRESH_ASSERT val 0x10
cde2 reg 0x 1c3      CD_SI1_SRC_AF_THRESH_NEGATE val 0x20
cde2 reg 0x 1c4              CD_SI1_SRC_CAL_LEN val 0x13
cde2 reg 0x 1c5             CD_SI1_SRC_CAL_ADDR val 0x13
cde2 reg 0x 1c6             CD_SI1_SRC_CAL_DATA val 0x8
cde2 reg 0x 1c7               CD_SI1_SNK_STATUS val 0x5
cde2 reg 0x 1c8               CD_SI1_SNK_CONFIG val 0x8
cde2 reg 0x 1c9      CD_SI1_SNK_AF_THRESH_ASSERT val 0x10
cde2 reg 0x 1ca      CD_SI1_SNK_AF_THRESH_NEGATE val 0x20
cde2 reg 0x 1cb              CD_SI1_SNK_CAL_LEN val 0x13
cde2 reg 0x 1cc             CD_SI1_SNK_CAL_ADDR val 0x13
cde2 reg 0x 1cd             CD_SI1_SNK_CAL_DATA val 0x8
cde2 reg 0x 1ce           CD_SI1_SNK_MISC_CONFIG val 0x11
cde2 reg 0x 1d7                  CD_SI1_SRC_INT val 0x0
cde2 reg 0x 1d8              CD_SI1_SRC_INT_MSK val 0xf
cde2 reg 0x 1d9               CD_SI1_SNK_INT__1 val 0x0
cde2 reg 0x 1da               CD_SI1_SNK_INT__0 val 0x0
cde2 reg 0x 1db           CD_SI1_SNK_INT_MSK__1 val 0xf
cde2 reg 0x 1dc           CD_SI1_SNK_INT_MSK__0 val 0xffff
cde2 reg 0x 100                   CD_XS2_XX_CFG val 0x0
cde2 reg 0x 101        CD_XS2_IXP1F_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 102        CD_XS2_IXP1B_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 103       CD_XS2_IXP1D0_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 104       CD_XS2_IXP1D1_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 105         CD_XS2_NTXF_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 106         CD_XS2_NTXS_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 107        CD_XS2_NTXD0_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 108        CD_XS2_NTXD1_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 109      CD_XS2_CC_IXP1_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 10a       CD_XS2_CC_NTX_ENQ_TRAP_CFG val 0x427
cde2 reg 0x 10b            CD_XS2_XX_ENQ_INT__2 val 0x0
cde2 reg 0x 10c            CD_XS2_XX_ENQ_INT__1 val 0x0
cde2 reg 0x 10d            CD_XS2_XX_ENQ_INT__0 val 0x0
cde2 reg 0x 10e         CD_XS2_XX_ENQ_INT_MSK__2 val 0xff
cde2 reg 0x 10f         CD_XS2_XX_ENQ_INT_MSK__1 val 0xffff
cde2 reg 0x 110         CD_XS2_XX_ENQ_INT_MSK__0 val 0xffff
cde2 reg 0x 111              CD_XS2_VOQ_PERR_INT val 0x0
cde2 reg 0x 112          CD_XS2_VOQ_PERR_INT_MSK val 0x3ff
cde2 reg 0x 113             CD_XS2_XX_VOQ_INT__1 val 0x0
cde2 reg 0x 114             CD_XS2_XX_VOQ_INT__0 val 0x0
cde2 reg 0x 115         CD_XS2_XX_VOQ_INT_MSK__1 val 0xf
cde2 reg 0x 116         CD_XS2_XX_VOQ_INT_MSK__0 val 0xffff
cde2 reg 0x 117                   CD_XS2_CDE2_INT val 0x0
cde2 reg 0x 118              CD_XS2_CDE2_INT_MSK val 0x7
cde2 reg 0x 119                CD_XS2_DI_HIT_INT val 0x0
cde2 reg 0x 11a            CD_XS2_DI_HIT_INT_MSK val 0x3ff
cde2 reg 0x 11b               CD_XS2_X2_PULL_STA val 0x3
cde2 reg 0x 11c                 CD_XS2_CC_FC_STA val 0x0
cde2 reg 0x 11d                CD_XS2_IX1_FC_STA val 0x0
cde2 reg 0x 11e                CD_XS2_NTX_FC_STA val 0x0
cde2 reg 0x 11f             CD_XS2_X2_VOQ_STA__1 val 0x3
cde2 reg 0x 120             CD_XS2_X2_VOQ_STA__0 val 0x303f
cde2 reg 0x 121                CD_XS2_XS2_GP_STA val 0x3
cde2 reg 0x 146    CD_XS2_NTX2CDE_HDRBYTE_HI_STA val 0x0
cde2 reg 0x 147    CD_XS2_NTX2CDE_HDRBYTE_LO_STA val 0x0
cde2 reg 0x 148    CD_XS2_CDE2NTX_HDRBYTE_HI_STA val 0x0
```

```
cde2 reg 0x 149    CD_XS2_CDE2NTX_HDRBYTE_LO_STA val 0x0
cde2 reg 0x 14a             CD_XS2_CD_XS2_FC_INT val 0x1000
cde2 reg 0x 14b         CD_XS2_CD_XS2_FC_INT_MSK val 0x0
cde2 reg 0x 14c            CD_XS2_CD_XS2_ERR_INT val 0x0
cde2 reg 0x 14d        CD_XS2_CD_XS2_ERR_INT_MSK val 0x7fff
```

Notes

As shown, statistics cover a range of CDE gates for the components in the ACE module. Values are indicated in hexadecimal number format. Also note keep in mind that there are actually two CDE units, cde1 and cde2.

Many of the following commands are related to this one and represent a subset of the information shown here. The descriptions for those commands may provide more information on specific fields.

show cde count

This command is a form of the **show cde** command that indicates whether multicast packets (generated by the CDE and reflected back by Hyperion) have been dropped by the Hyperion receive registers (HR). The Hyperion ASIC is the packet rewrite, multicast, and SPAN engine used by the ACE to receive connections over the Cat6k Switching backplane.

Sample Output

```
switch/Admin#  show cde count
CDE1 reg 0x1f CD_HR_INT1 bit 8 count 136506 time 68275173
[MC_FILTER_DROP]
```

Notes

In the sample output, the count indicates that packets have been filtered by the CDE. This is not unusual. In this case, a component of the ACE has generated a multicast packet which was sent through the CDE to the Hyperion ASIC that serves as the interface to the Cat6k switching backplane. The Hyperion ASIC, recognizing the message as a multicast message, floods it. The Hyperion receive registers on the CDE drop the packet.

show cde health

For purposes of general troubleshooting, this command is the most useful of the CDE-related show commands. It can be the best place to start to inspect the internal status of the ACE Module infrastructure.

Its output describes the CDE ports, the interfaces between the CDE and all other components of the ACE Module.

Sample Output

```
switch/Admin# show cde health

CDE BRCM INTERFACE
======================
Packets received                                        122503
Packets transmitted                                     352125
```

```
Broadcom interface CRC error count                                0
BRCM VOQ status                          [empty]      [not full]
BRCM pull status                                    [not pulling]

CDE HYPERION INTERFACE
======================
Packets received                                         10777393
Packets transmitted                                       7112980
Short packets drop count                                        0
Fifo Full drop count                                            0
Protocol error drop count                                       0
FCS error drop count                                            0
CRC error drop count                                            0
Num times flow control triggered on hyp interface               0
Num self generated multicast packets filtered             6942228
HYP IXP0 VOQ status                      [empty]      [not full]
HYP IXP1 VOQ status                      [empty]      [not full]
HYP SLOW VOQ status                      [empty]      [not full]
HYP tx pull status                                     [pulling]

CDE IXP0 INTERFACE
======================
Packets received                                          7064084
Packets transmitted                                       3347755
Num bad pkts recvd on fast spi channel0                         0
Num bad pkts recvd on slow spi channel8                         0
Num bad pkts recvd on fast spi channel2                         0
Num bad pkts recvd on slow spi channel4                         0
IXP0 Fast VOQ status                     [empty]      [not full]
IXP0 BRCM VOQ status                     [empty]      [not full]
IXP0 pull status                                       [pulling]
IXP0 spi src status                                    [healthy]
IXP0 spi snk status                                    [healthy]

CDE1 SWITCH1 INTERFACE
======================
Packets received (hyp, ixp0)                               195814
Packets received (bcm)                                     205207
Packets received (daughter card 0)                              0
Packets received (daughter card 1)                              0
Packets Errors received (hyp, ixp0)                             0
Packets Errors received (bcm)                                   0
Packets Errors received (daughter card 0)                       0
Packets Errors received (daughter card 1)                       0
Packets transmitted (ixp1)                                 609913
Packets transmitted (nitrox)                                    0
Packets Errors transmitted (ixp1)                               0
Packets Errors transmitted (nitrox)                             0

CDE2 SWITCH2 INTERFACE
======================
Packets received (ixp1)                                    609913
Packets received (nitrox)                                       0
Packets Errors received (ixp1)                                  0
Packets Errors received (nitrox)                                0
Packets transmitted (hyp, ixp0)                            195814
Packets transmitted (broadcom)                             205207
Packets transmitted (daughter card 0)                           0
Packets transmitted (daughter card 1)                           0
Packets Errors transmitted (ixp1)                               0
Packets Errors transmitted (nitrox)                             0
Packets Errors transmitted (daughter card 0)                    0
Packets Errors transmitted (daughter card 1)                    0
```

```
CDE IXP1 INTERFACE
======================
Packets received                                          401021
Packets transmitted                                       609913
Num bad pkts recvd on fast spi channel0                        0
Num bad pkts recvd on slow spi channel8                        0
Num bad pkts recvd on fast spi channel2                        0
Num bad pkts recvd on slow spi channel4                        0
IXP1 Fast VOQ status                        [empty]      [not full]
IXP1 BRCM VOQ status                        [empty]      [not full]
IXP1 pull status                                          [pulling]
IXP1 spi src status                                       [healthy]
IXP1 spi snk status                                       [healthy]

CDE NITROX INTERFACE
======================
Packets received                                               0
Packets transmitted                                            0
Num bad pkts recvd on fast spi channel0                        0
Num bad pkts recvd on slow spi channel8                        0
Num bad pkts recvd on fast spi channel2                        0
Num bad pkts recvd on slow spi channel4                        0
NTX Fast VOQ status                         [empty]      [not full]
NTX BRCM VOQ status                         [empty]      [not full]
NTX pull status                                           [pulling]
NTX spi src status                                        [healthy]
NTX spi snk status                                        [healthy]
```

Notes

The labeled components are:

- BRCM is the CP (Control Processor) for the module
- HYPERION is the ASIC from which/to which the CDE receives/sends data traffic.
- IXP0 and IXP1 are the two network processors
- NITROX is the SSL hardware decrypt/encrypt chip.

In general, the components should:

1. show a "healthy" status for spi src/snk.
2. show "pulling" for "component pull status" (the exception being the BRCM, which may show "not pulling" even when pulling. (There are CDE registers which can be used to find out if this is problematic condition or just a false align.)
3. show their queues as "empty"/"not full".
4. have incrementing packet receive/send counters.

Conditions other than listed would be considered unusual and warrant further investigation.

show cde int

A form of the **show cde** command that shows only the CDE interrupts and masks. This is information on the internal operation of the ACE Module hardware and in general useful only to internal development.

Sample Output

```
switch/Admin# show cde int
cde1 reg 0x   2                    CD_CP_ERR_INT val 0x0
cde1 reg 0x   3                CD_CP_ERR_INT_MSK val 0x3
cde1 reg 0x  1d                       CD_HR_INT0 val 0x0
cde1 reg 0x  1e                   CD_HR_INT0_MSK val 0xffff
cde1 reg 0x  1f                       CD_HR_INT1 val 0x100
cde1 reg 0x  20                   CD_HR_INT1_MSK val 0x2ff
cde1 reg 0x  ca                        CD_HT_INT val 0x0
cde1 reg 0x  cb                    CD_HT_INT_MSK val 0x7ff
cde1 reg 0x 197                   CD_SI0_SRC_INT val 0x0
cde1 reg 0x 198               CD_SI0_SRC_INT_MSK val 0xf
cde1 reg 0x 199                CD_SI0_SNK_INT__1 val 0x0
cde1 reg 0x 19a                CD_SI0_SNK_INT__0 val 0x0
cde1 reg 0x 19b            CD_SI0_SNK_INT_MSK__1 val 0xf
cde1 reg 0x 19c            CD_SI0_SNK_INT_MSK__0 val 0xffff
cde1 reg 0x 104            CD_XS1_DI_CRC_ERR_INT val 0x0
cde1 reg 0x 105        CD_XS1_DI_CRC_ERR_INT_MSK val 0x3
cde1 reg 0x 106             CD_XS1_XX_ENQ_INT__3 val 0x0
cde1 reg 0x 107             CD_XS1_XX_ENQ_INT__2 val 0x0
cde1 reg 0x 108             CD_XS1_XX_ENQ_INT__1 val 0x0
cde1 reg 0x 109             CD_XS1_XX_ENQ_INT__0 val 0x0
cde1 reg 0x 10a         CD_XS1_XX_ENQ_INT_MSK__3 val 0xff
cde1 reg 0x 10b         CD_XS1_XX_ENQ_INT_MSK__2 val 0xffff
cde1 reg 0x 10c         CD_XS1_XX_ENQ_INT_MSK__1 val 0xffff
cde1 reg 0x 10d         CD_XS1_XX_ENQ_INT_MSK__0 val 0xffff
cde1 reg 0x 10e             CD_XS1_XX_VOQ_INT__2 val 0x0
cde1 reg 0x 10f             CD_XS1_XX_VOQ_INT__1 val 0x0
cde1 reg 0x 110             CD_XS1_XX_VOQ_INT__0 val 0x0
cde1 reg 0x 111         CD_XS1_XX_VOQ_INT_MSK__2 val 0x3
cde1 reg 0x 112         CD_XS1_XX_VOQ_INT_MSK__1 val 0xffff
cde1 reg 0x 113         CD_XS1_XX_VOQ_INT_MSK__0 val 0xffff
cde1 reg 0x 114        CD_XS1_XX_VOQ_PERR_INT__1 val 0x0
cde1 reg 0x 115        CD_XS1_XX_VOQ_PERR_INT__0 val 0x0
cde1 reg 0x 116    CD_XS1_XX_VOQ_PERR_INT_MSK__1 val 0x1
cde1 reg 0x 117    CD_XS1_XX_VOQ_PERR_INT_MSK__0 val 0xffff
cde1 reg 0x 118            CD_XS1_XX_DI_TRAP_INT val 0x0
cde1 reg 0x 119        CD_XS1_XX_DI_TRAP_INT_MSK val 0x3fff
cde1 reg 0x 11a                    CD_XS1_CC_INT val 0x0
cde1 reg 0x 11b                CD_XS1_CC_INT_MSK val 0x3
cde1 reg 0x 11c            CD_XS1_BI_CRC_ERR_INT val 0x0
cde1 reg 0x 11d        CD_XS1_BI_CRC_ERR_INT_MSK val 0x1
cde1 reg 0x 151          CD_XS1_CD_XS1_FC_INT__1 val 0x0
cde1 reg 0x 152          CD_XS1_CD_XS1_FC_INT__0 val 0x0
cde1 reg 0x 153      CD_XS1_CD_XS1_FC_INT_MSK__1 val 0x0
cde1 reg 0x 154      CD_XS1_CD_XS1_FC_INT_MSK__0 val 0x0
cde2 reg 0x   2                    CD_CP_ERR_INT val 0x0
cde2 reg 0x   3                CD_CP_ERR_INT_MSK val 0x3
cde2 reg 0x 197                   CD_SI0_SRC_INT val 0x0
cde2 reg 0x 198               CD_SI0_SRC_INT_MSK val 0xf
cde2 reg 0x 199                CD_SI0_SNK_INT__1 val 0x0
cde2 reg 0x 19a                CD_SI0_SNK_INT__0 val 0x0
cde2 reg 0x 19b            CD_SI0_SNK_INT_MSK__1 val 0xf
cde2 reg 0x 19c            CD_SI0_SNK_INT_MSK__0 val 0xffff
cde2 reg 0x 1d7                   CD_SI1_SRC_INT val 0x0
cde2 reg 0x 1d8               CD_SI1_SRC_INT_MSK val 0xf
cde2 reg 0x 1d9                CD_SI1_SNK_INT__1 val 0x0
cde2 reg 0x 1da                CD_SI1_SNK_INT__0 val 0x0
cde2 reg 0x 1db            CD_SI1_SNK_INT_MSK__1 val 0xf
cde2 reg 0x 1dc            CD_SI1_SNK_INT_MSK__0 val 0xffff
cde2 reg 0x 10b             CD_XS2_XX_ENQ_INT__2 val 0x0
cde2 reg 0x 10c             CD_XS2_XX_ENQ_INT__1 val 0x0
cde2 reg 0x 10d             CD_XS2_XX_ENQ_INT__0 val 0x0
cde2 reg 0x 10e         CD_XS2_XX_ENQ_INT_MSK__2 val 0xff
cde2 reg 0x 10f         CD_XS2_XX_ENQ_INT_MSK__1 val 0xffff
```

```
cde2 reg 0x 110         CD_XS2_XX_ENQ_INT_MSK__0 val 0xffff
cde2 reg 0x 111              CD_XS2_VOQ_PERR_INT val 0x0
cde2 reg 0x 112          CD_XS2_VOQ_PERR_INT_MSK val 0x3ff
cde2 reg 0x 113             CD_XS2_XX_VOQ_INT__1 val 0x0
cde2 reg 0x 114             CD_XS2_XX_VOQ_INT__0 val 0x0
cde2 reg 0x 115         CD_XS2_XX_VOQ_INT_MSK__1 val 0xf
cde2 reg 0x 116         CD_XS2_XX_VOQ_INT_MSK__0 val 0xffff
cde2 reg 0x 117                   CD_XS2_CDE2_INT val 0x0
cde2 reg 0x 118              CD_XS2_CDE2_INT_MSK val 0x7
cde2 reg 0x 119                CD_XS2_DI_HIT_INT val 0x0
cde2 reg 0x 11a             CD_XS2_DI_HIT_INT_MSK val 0x3ff
cde2 reg 0x 14a             CD_XS2_CD_XS2_FC_INT val 0x1000
cde2 reg 0x 14b         CD_XS2_CD_XS2_FC_INT_MSK val 0x0
cde2 reg 0x 14c            CD_XS2_CD_XS2_ERR_INT val 0x0
cde2 reg 0x 14d        CD_XS2_CD_XS2_ERR_INT_MSK val 0x7fff
```

show cde stat delta

For the CDE statistics that indicate packets, transmits, and errors, this command shows the value differences since the previous time the command was run. This is primarily useful for internal development to learn how traffic is moving through the module.

To use the command, you would typically run it several times to see how traffic is affecting the statistics for particular registers.

Sample Output

```
switch/Admin# sho cde stats delta
cde1 reg 0x  18                 CD_HR_PKT_CNT__1 val 0
cde1 reg 0x  19                 CD_HR_PKT_CNT__0 val 776
cde1 reg 0x  1a         CD_HR_SHORT_PKT_DROP_CNT val 0
cde1 reg 0x  1b          CD_HR_FULL_PKT_DROP_CNT val 0
cde1 reg 0x  1c           CD_HR_PKT_PROT_ERR_CNT val 0
cde1 reg 0x  21                CD_HR_FCS_ERR_CNT val 0
cde1 reg 0x  22                CD_HR_CRC_ERR_CNT val 0
cde1 reg 0x  23              CD_HR_FC_EVENT_CNT val 0
cde1 reg 0x  24         CD_HR_MC_FILTER_DROP_CNT val 307
cde1 reg 0x  c8                  CD_HT_TX_CNT__1 val 0
cde1 reg 0x  c9                  CD_HT_TX_CNT__0 val 324
cde1 reg 0x  cc                  CD_HT_TX_RX_CNT val 324
cde1 reg 0x 18f            CD_SI0_SRC_PKT_CNT__1 val 0
cde1 reg 0x 190            CD_SI0_SRC_PKT_CNT__0 val 446
cde1 reg 0x 191            CD_SI0_SNK_PKT_CNT__1 val 0
cde1 reg 0x 192            CD_SI0_SNK_PKT_CNT__0 val 314
cde1 reg 0x 193           CD_SI0_SNK_CH0_ERR_CNT val 0
cde1 reg 0x 194           CD_SI0_SNK_CH8_ERR_CNT val 0
cde1 reg 0x 195           CD_SI0_SNK_CH2_ERR_CNT val 0
cde1 reg 0x 196           CD_SI0_SNK_CH4_ERR_CNT val 0
cde1 reg 0x 128           CD_XS1_HYP_ENQ_PKTCNT val 469
cde1 reg 0x 129         CD_XS1_IXP0F_ENQ_PKTCNT val 293
cde1 reg 0x 12a        CD_XS1_IXP0BCM_ENQ_PKTCNT val 21
cde1 reg 0x 12b         CD_XS1_IXP0D0_ENQ_PKTCNT val 0
cde1 reg 0x 12c         CD_XS1_IXP0D1_ENQ_PKTCNT val 0
cde1 reg 0x 12d            CD_XS1_BCM_ENQ_PKTCNT val 44
cde1 reg 0x 12e           CD_XS1_DB0F_ENQ_PKTCNT val 0
cde1 reg 0x 12f           CD_XS1_DB0S_ENQ_PKTCNT val 0
cde1 reg 0x 130           CD_XS1_DB1F_ENQ_PKTCNT val 0
cde1 reg 0x 131           CD_XS1_DB1S_ENQ_PKTCNT val 0
cde1 reg 0x 132        CD_XS1_CC2FAST_ENQ_PKTCNT val 31
```

```
cde1 reg 0x 133         CD_XS1_CC2BCM_ENQ_PKTCNT val 29
cde1 reg 0x 134          CD_XS1_CC2D0_ENQ_PKTCNT val 0
cde1 reg 0x 135          CD_XS1_CC2D1_ENQ_PKTCNT val 0
cde1 reg 0x 136        CD_XS1_CC_RX_PKT_CNT_FAST val 31
cde1 reg 0x 137         CD_XS1_CC_RX_PKT_CNT_BCM val 29
cde1 reg 0x 138          CD_XS1_CC_RX_PKT_CNT_D0 val 0
cde1 reg 0x 139          CD_XS1_CC_RX_PKT_CNT_D1 val 0
cde1 reg 0x 13a        CD_XS1_CC_RX_ERR_CNT_FAST val 0
cde1 reg 0x 13b         CD_XS1_CC_RX_ERR_CNT_BCM val 0
cde1 reg 0x 13c          CD_XS1_CC_RX_ERR_CNT_D0 val 0
cde1 reg 0x 13d          CD_XS1_CC_RX_ERR_CNT_D1 val 0
cde1 reg 0x 13e        CD_XS1_CC_TX_PKT_CNT_IXP1 val 67
cde1 reg 0x 13f      CD_XS1_CC_TX_PKT_CNT_NITROX val 0
cde1 reg 0x 140        CD_XS1_CC_TX_ERR_CNT_IXP1 val 0
cde1 reg 0x 141      CD_XS1_CC_TX_ERR_CNT_NITROX val 0
cde1 reg 0x 142               CD_XS1_BI_RXCNT__1 val 0
cde1 reg 0x 143               CD_XS1_BI_RXCNT__0 val 44
cde1 reg 0x 144               CD_XS1_BI_TXCNT__1 val 0
cde1 reg 0x 145               CD_XS1_BI_TXCNT__0 val 50
cde1 reg 0x 146             CD_XS1_DB0_TXCNT__1 val 0
cde1 reg 0x 147             CD_XS1_DB0_TXCNT__0 val 0
cde1 reg 0x 148             CD_XS1_DB0_RXCNT__1 val 0
cde1 reg 0x 149             CD_XS1_DB0_RXCNT__0 val 0
cde1 reg 0x 14a           CD_XS1_DB0_RX_CRC_CNT val 0
cde1 reg 0x 14b             CD_XS1_DB1_TXCNT__1 val 0
cde1 reg 0x 14c             CD_XS1_DB1_TXCNT__0 val 0
cde1 reg 0x 14d             CD_XS1_DB1_RXCNT__1 val 0
cde1 reg 0x 14e             CD_XS1_DB1_RXCNT__0 val 0
cde1 reg 0x 14f           CD_XS1_DB1_RX_CRC_CNT val 0
cde1 reg 0x 150           CD_XS1_BI_CRC_ERR_CNT val 0
cde2 reg 0x 18f           CD_SI0_SRC_PKT_CNT__1 val 0
cde2 reg 0x 190           CD_SI0_SRC_PKT_CNT__0 val 0
cde2 reg 0x 191           CD_SI0_SNK_PKT_CNT__1 val 0
cde2 reg 0x 192           CD_SI0_SNK_PKT_CNT__0 val 0
cde2 reg 0x 193         CD_SI0_SNK_CH0_ERR_CNT val 0
cde2 reg 0x 194         CD_SI0_SNK_CH8_ERR_CNT val 0
cde2 reg 0x 195         CD_SI0_SNK_CH2_ERR_CNT val 0
cde2 reg 0x 196         CD_SI0_SNK_CH4_ERR_CNT val 0
cde2 reg 0x 1cf          CD_SI1_SRC_PKT_CNT__1 val 0
cde2 reg 0x 1d0          CD_SI1_SRC_PKT_CNT__0 val 67
cde2 reg 0x 1d1          CD_SI1_SNK_PKT_CNT__1 val 0
cde2 reg 0x 1d2          CD_SI1_SNK_PKT_CNT__0 val 60
cde2 reg 0x 1d3         CD_SI1_SNK_CH0_ERR_CNT val 0
cde2 reg 0x 1d4         CD_SI1_SNK_CH8_ERR_CNT val 0
cde2 reg 0x 1d5         CD_SI1_SNK_CH2_ERR_CNT val 0
cde2 reg 0x 1d6         CD_SI1_SNK_CH4_ERR_CNT val 0
cde2 reg 0x 122         CD_XS2_IXP1F_ENQ_PKTCNT val 31
cde2 reg 0x 123         CD_XS2_IXP1B_ENQ_PKTCNT val 29
cde2 reg 0x 124        CD_XS2_IXP1D0_ENQ_PKTCNT val 0
cde2 reg 0x 125        CD_XS2_IXP1D1_ENQ_PKTCNT val 0
cde2 reg 0x 126          CD_XS2_NTXF_ENQ_PKTCNT val 0
cde2 reg 0x 127          CD_XS2_NTXS_ENQ_PKTCNT val 0
cde2 reg 0x 128        CD_XS2_NTXD0_ENQ_PKTCNT val 0
cde2 reg 0x 129        CD_XS2_NTXD1_ENQ_PKTCNT val 0
cde2 reg 0x 12a        CD_XS2_CC_RX_PKT_CNT_IXP1 val 67
cde2 reg 0x 12b         CD_XS2_CC_RX_PKT_CNT_NTX val 0
cde2 reg 0x 12c        CD_XS2_CC_RX_ERR_CNT_IXP1 val 0
cde2 reg 0x 12d         CD_XS2_CC_RX_ERR_CNT_NTX val 0
cde2 reg 0x 12e        CD_XS2_CC_TX_PKT_CNT_FAST val 31
cde2 reg 0x 12f         CD_XS2_CC_TX_PKT_CNT_BCM val 29
cde2 reg 0x 130          CD_XS2_CC_TX_PKT_CNT_D0 val 0
cde2 reg 0x 131          CD_XS2_CC_TX_PKT_CNT_D1 val 0
cde2 reg 0x 132        CD_XS2_CC_TX_ERR_CNT_FAST val 0
cde2 reg 0x 133         CD_XS2_CC_TX_ERR_CNT_BCM val 0
```

```
cde2 reg 0x 134          CD_XS2_CC_TX_ERR_CNT_D0 val 0
cde2 reg 0x 135          CD_XS2_CC_TX_ERR_CNT_D1 val 0
cde2 reg 0x 136     CD_XS2_CDE2NTX_CH8_PKTCNT__1 val 0
cde2 reg 0x 137     CD_XS2_CDE2NTX_CH8_PKTCNT__0 val 0
cde2 reg 0x 138     CD_XS2_CDE2NTX_CH4_PKTCNT__1 val 0
cde2 reg 0x 139     CD_XS2_CDE2NTX_CH4_PKTCNT__0 val 0
cde2 reg 0x 13a     CD_XS2_CDE2NTX_CH2_PKTCNT__1 val 0
cde2 reg 0x 13b     CD_XS2_CDE2NTX_CH2_PKTCNT__0 val 0
cde2 reg 0x 13c     CD_XS2_CDE2NTX_CH0_PKTCNT__1 val 0
cde2 reg 0x 13d     CD_XS2_CDE2NTX_CH0_PKTCNT__0 val 0
cde2 reg 0x 13e     CD_XS2_NTX2CDE_CH8_PKTCNT__1 val 0
cde2 reg 0x 13f     CD_XS2_NTX2CDE_CH8_PKTCNT__0 val 0
cde2 reg 0x 140     CD_XS2_NTX2CDE_CH4_PKTCNT__1 val 0
cde2 reg 0x 141     CD_XS2_NTX2CDE_CH4_PKTCNT__0 val 0
cde2 reg 0x 142     CD_XS2_NTX2CDE_CH2_PKTCNT__1 val 0
cde2 reg 0x 143     CD_XS2_NTX2CDE_CH2_PKTCNT__0 val 0
cde2 reg 0x 144     CD_XS2_NTX2CDE_CH0_PKTCNT__1 val 0
cde2 reg 0x 145     CD_XS2_NTX2CDE_CH0_PKTCNT__0 val 0
```

show cfgmgr internal table rserver

Displays the internal ID for the real servers (rservers) configured in the active context. This information is useful for performing certain troubleshooting tasks, such as using LbInspectTool. LbInspectTool is a software tool used by Cisco support personnel to inspect data structures associated with the loadbalancing process that runs on the dataplane.

Sample Output

```
ACE30001/Admin# show cfgmgr internal table rserver

Rserver-id  Rserver-Name      Ctx Id  Encap Flags
-------------------------------------------------------------------
1           SSG1               0      4     ADDED, UPDATED, RELOADED, DATA_VALID,
2           SSG2               0      49    ADDED, UPDATED, RELOADED, DATA_VALID,
3           SSG3               0      3     ADDED, UPDATED, RELOADED, DATA_VALID,
```

Notes

The internal ID number for each real server is shown in the Rserver-id column. This is the value that the LbInspectTool command takes as input to produce information for that rserver:

```
'1 - Inspect tables ' / 'd' - DRAM / 'R' - Rserver / 'Enter ID: '
```

show cfgmgr internal table sfarm

Displays the internal ID for the server farms configured in the active context. This information is useful for performing certain troubleshooting tasks, such as using LbInspectTool.

Sample Output

```
ACE30001/rlb_ssg# show cfgmgr internal table sfarm
```

```
Sfarm-Id   Sfarm-name                 Sfarm-Type Ctx-Id  Flags
-------------------------------------------------------------------------
4          SSG_RLB                   Host       2      DATA_VALID,
5          Second_serverfarm         Host       2      DATA_VALID,
```

Notes

The internal ID number for each server farm is shown in the Sfarm-Id column. This is the value that the LbInspectTool command takes as input to produce information for that server farm:

```
'1 - Inspect tables ' / 'd' - DRAM / 's' - Server Farm / 'Enter ID: '
```

show cfgmgr internal table sfarm-real

Lists the real servers along with their associated server farm in the context. This command differs from the **show cfgmgr internal table rserver** command in that **show cfgmgr internal table rserver** displays one index for each physical rserver, whereas this command displays an entry for each occurrence of the rserver in a serverfarm.

In other words, a real server that is in multiple server farms will be shown multiple times.

Sample Output

```
ACE30001/rlb_ssg# show cfgmgr internal table sfarm-real

Real-Id  Real-Server-Name Port   Sfarm-Name        Ctx-Id  Flags
-------------------------------------------------------------------------
7        SSG1             0      SSG_RLB            2      DATA_VALID,
8        SSG2             0      SSG_RLB            2      DATA_VALID,
9        SSG3             0      SSG_RLB            2      DATA_VALID,
10       FC8-server       0      https-test         2      DATA_VALID,
15       SSG1             0      Second_serverfarm  2      ADDED, UPDATED, DATA_VALID,
16       SSG2             0      Second_serverfarm  2      ADDED, UPDATED, DATA_VALID,
17       SSG3             0      Second_serverfarm  2      ADDED, UPDATED, DATA_VALID,
```

show conn

Displays the connections currently being handled by ACE.

Sample Output

```
ACE30001/Admin# show conn display 1000 detail`

total current connections : 143930
display first num connection pairs : 1000

conn-id    np dir proto vlan source                destination           state
----------+--+---+-----+----+---------------------+---------------------+------+
9          1  in  TCP   101  209.165.201.20:80       10.87.3.184:34484     CLOSED
          [ idle time   : 00:01:13,   byte count  : 40         ]
          [ elapsed time: 00:47:59,   packet count: 1          ]
```

```
8          1  out TCP   203  10.87.3.184:34484     209.165.201.20:80       ESTAB
          [ conn in reuse pool : FALSE]
          [ idle time   : 00:01:13,   byte count  : 0          ]
          [ elapsed time: 00:47:59,   packet count: 0          ]
89867      1  in  TCP   101  209.165.201.20:80       10.87.3.188:54331     ESTAB
          [ idle time   : 00:01:13,   byte count  : 40         ]
          [ elapsed time: 00:11:01,   packet count: 1          ]
10         1  out TCP   203  10.87.3.188:54331     209.165.201.20:80       ESTAB
          [ conn in reuse pool : FALSE]
          [ idle time   : 00:01:13,   byte count  : 0          ]
          [ elapsed time: 00:11:01,   packet count: 0          ]
12         1  in  UDP   105  192.168.5.179:50000   192.168.5.166:50002   --
--         -  -   --    --   --                    --                    --
```

Notes

Resource	Maximum Value
conn-id	The unique identifier for the connection.
np	The IXP handling this connection.
dir	The direction of the connection, from the perspective of the ACE (in or out).
proto	The TCP/IP protocol for this connection.
vlan	The VLAN used for this connection.
Source	The source IP address and port number.
destination	The destination IP address and port number.
state	The state of the connection. Non-TCP connections display "--" Possible TCP states are: • INIT ? Initial state of a connection. • SYNSEEN ? ACE received a SYN. • SYNACK ? ACE sent a SYNACK. • ESTAB ? 3-way handshake completed and connection is established. • CLSFIN ? ACE closed the connection with a FIN. • CLSRST ? ACE closed the connection with a RST. • CLSTIMEOUT ? ACE closed the connection for it timed out. • CLOSED ? Connection is half closed.

show context

Sample Output

```
ACE5/Admin# show context

Number of Contexts = 1

Name: Admin , Id: 0
Config count: 213
Description:
Resource-class: gold
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled
```

Notes

For troubleshooting purposes, the only statistic of interest is "Config count". It is the number of successful configuration commands.

show crypto authgroup

This command may be called with the "all" keyword or the name of a configured chain group.

Sample Output

```
ACE30001/Admin# show crypto authgroup all
authgroup TestAuthgroup contains:
  MyRootCA
  MySubCA1

ACE30001/Admin# show crypto authgroup TestAuthgroup
authgroup TestAuthgroup contains:

MyRootCA:
  Subject: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com
  Issuer: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com

MySubCA1:
  Subject: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=SubCA1/CN=MySubCA1
  Issuer: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com
```

Notes

An authgroup specifies the list of CA certificates that the ACE will use to authenticate its peer certificate. This is required if SSL initiation is configured (to authenticate the server), or if ACE is configured as an SSL server and client authentication is configured.

The fields displayed in this command are, for each certificate in the authgroup:

- Subject: The distinguished name of the organization that owns the certificate and possesses the private key.
- Issuer: The distinguished name of the CA that issued the cert.

show crypto cdp-errors

Sample Output

```
ACE30001/Admin# show crypto cdp-errors

 Incomplete:                0        Malformed:                 0
 Unrecognized Transports:   0        Missing from cert:         0
 Best Effort CDP Errors Ignored: 0
```

Notes

A CDP is a CRL distribution point. This command lists the number of times various errors were encountered when trying to parse the CDP.

The counters and the reasons they are incremented are:

Counter	Reason
Incomplete	There is no '/' nor ':' in the CDP. There is no hostname in the CDP filename or base and attributes not provided. If you enable SSL error debugging, you will see this message.
Improper length of the filename or base and attrs	If you enable SSL error debugging, you will see this message; could not find "certificateRevocationList" or "certificateRevocationList;binary" in the LDAP URI; LDAP URI scope is not "one", "base", or "sub" filter in url NOT cRLDistributionPoint or wrong format; if you enable SSL error debugging, you will see this message; something wrong in the URL, ignored (the LDAP URI is > 255 characters); if you enable SSL error debugging, you will see this message.
Unrecognized Transports:	CDP does not start with "http://" or "ldap://"; if you enable SSL error debugging, you will see a message.
Missing from cert	received cdp missing indication from DP; if you enable SSL error debugging, you will see this message.
Best Effort CDP Errors Ignored	The revocation of the cert needed to be reverted based on CDP issues found in the cert. This is applicable for best effort CRLs only (A3(2.x) and later)

show crypto certificate

With the **all** keyword, this command shows summary information of all certificates in the context. When you indicate a specific certificate in the command, it shows details for that certificate.

Sample Output

```
ace19/Admin# sho crypto certificate all

All Certificate Files Loaded:
cisco-sample-cert:
Subject: /C=IN/ST=KA/L=BLR/O=CISCO/OU=ADBU/CN=SSL-TEST
Issuer: /C=IN/ST=KA/L=BLR/O=CISCO/OU=ADBU/CN=SSL-TEST
Not Before: Apr  3 09:50:55 2009 GMT
Not After: Apr  1 09:50:55 2019 GMT
CA Cert: TRUE


ace19/Admin# show crypto certificate cisco-sample-cert
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number:
           ad:e4:e2:f1:50:b7:ce:bd
       Signature Algorithm: sha1WithRSAEncryption
       Issuer: C=IN, ST=KA, L=BLR, O=CISCO, OU=ADBU, CN=SSL-TEST
       Validity
           Not Before: Apr  3 09:50:55 2009 GMT
           Not After : Apr  1 09:50:55 2019 GMT
       Subject: C=IN, ST=KA, L=BLR, O=CISCO, OU=ADBU, CN=SSL-TEST
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
```

```
                Modulus (1024 bit):
                    00:cf:a2:60:66:5b:ce:b6:38:6f:94:df:0d:1c:61:
                    41:6c:48:82:23:e6:6b:86:01:22:3a:f7:9a:a4:60:
                    5e:b2:5a:50:5d:40:ca:9a:9a:13:b1:8b:16:95:9a:
                    26:af:7a:05:49:ed:8d:93:3b
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                A1:7A:E2:50:54:9D:82:86:A5:01:F5:14:7B:78:0D:AE:
                12:18:0C:D9
            X509v3 Authority Key Identifier:
                keyid:A1:7A:E2:50:54:9D:82:86:A5:01:F5:14:7B:78:0D:
                   AE:12:18:0C:D9
                DirName:/C=IN/ST=KA/L=BLR/O=CISCO/OU=ADBU/CN=SSL-TEST
                serial:AD:E4:E2:F1:50:B7:CE:BD

            X509v3 Basic Constraints:
                CA:TRUE
    Signature Algorithm: sha1WithRSAEncryption
        7c:ae:3a:96:03:e9:86:e8:40:6f:d4:d1:2a:88:fd:b5:60:7a:
        90:07:e2:de:9e:99:b8:e9:1e:f4:aa:c1:b6:16:0a:df:a1:d6:
        b8:73:12:08:b4:33:ba:21:7b:97:60:4c:1c:d1:a2:cd:e0:dd:
        99:84:56:c1:13:91:28:86:6f:89:30:b0:0e:96:fc:a0:d1:92:
        c4:7d:44:03:0b:93:0a:6f:40:67:99:ce:a1:1c:d4:5f:40:a2:
        f9:e0
```

Notes

Field	Description
Subject	The distinguished name of the organization that owns the certificate and possesses the private key.
Issuer	The distinguished name of the CA that issued the cert.
Not Before	Starting time period before which the certificate is not considered valid.
Not After	Ending time period after which the certificate is not considered valid.
CA Cert	Indicates whether this cert belongs to a Certificate Authority.

show crypto chaingroup

Shows the certificates in a specified chaingroup or all chain groups. A chain group specifies the list of certificate that the ACE sends to its peer during the handshake. A certificate chain is a hierarchical list of certificates that includes the subject?s certificate, the root CA certificate, and any intermediate CA certificates in between.

Sample Output

```
ACE30001/Admin# show crypto chaingroup TestChaingroup
chaingroup TestChaingroup contains:

MyRootCA:
  Subject: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com
  Issuer: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com

MySubCA1:
  Subject: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=SubCA1/CN=MySubCA1
```

```
Issuer: /C=US/ST=MA/L=Boxborough/O=MyOrg/OU=Root/CN=MyRootCA/emailAddress=user@example.com
```

Notes

Field	Description
Subject	The distinguished name of the organization that owns the certificate and possesses the private key.
Issuer	The distinguished name of the CA that issued the certificate.

show crypto crl

Shows the certificate revocation lists downloaded to the device.

Sample Output

```
switch/Admin# show crypto crl crl3
crl3:
URL: ldap://ex55.example.com/CN=ACE-NAREN-ROOT-CA(2),CN=cis
co-5jbtgrx93,CN=CDP,CN=Public%20Key%20Services,CN=Services,CN=Configuration,DC=s
slvpn-ex55,DC=cisco,DC=com?certificateRevocationList?base?objectclass=cRLDistrib
utionPoint
Last Downloaded: Tue Nov 25 09:31:31 2009
```

show crypto csr-params

Shows the values of a particular Certificate Signing Request (CSR) parameter set.

Sample Output

```
ACE30001/Admin# show crypto csr-params TestCsr
        country-name:   US
        state:          MA
        locality:       Boxborough
        org-name:       Cisco
        org-unit:       ADBU
        common-name:    TestCA
        serial-number:  00010203040506
        email:          testAdmin@example.com
```

Notes

Field	Description
Country-name	Country where the certificate owner resides.
State	State where the certificate owner resides.
Locality	Locality where the certificate owner resides.
Org-name	Name of the organization (certificate owner or subject).
Org-unit	Name of unit within the organization.
Common-name	Common-name (domain name or individual hostname of the SSL site).

Serial number	The serial number assigned by the CA to the certificate and which is unique for certificates issued by that particular CA.
Email	E-mail address of the certificate owner.

show crypto files

Lists the SSL files loaded on the ACE, including the preloaded sample key and cert.

Sample Output

```
ACE30001/Admin# show crypto files
Filename                                  File  File    Expor      Key/
                                          Size  Type    table      Cert
------------------------------------------------------------------------
cisco-sample-cert                         1082  PEM     Yes        CERT
cisco-sample-key                          887   PEM     Yes         KEY
NamedRootCA_server1_cert.pem              2978  PEM     Yes        CERT
NamedRootCA_server1_key.pem               963   PEM     Yes         KEY
```

Notes

Field	Description
File Size	Size of the file in bytes.
Exportable	Indicates whether you can export the file from the ACE using the crypto export command: If 'Yes', then you can export the file to an FTP, SFTP, or TFP server
File Type	Format of the file: PEM, DER, or PKCS12
Key/Cert	Indicates whether the file is a certificate or a private key, or

show crypto key

With the **all** keyword, this command lists a summary of all keys stored on the context. When you indicate a specific key, the command shows details for that key.

Sample Output

```
ace19/Admin#  show crypto key all
Filename                                  Bit Size Type
--------                                  -------- ----
cisco-sample-key                          1024     RSA

ace19/Admin#  show crypto key cisco-sample-key

1024 bit RSA keypair found in cisco-sample-key
Modulus:
cf:a2:60:66:5b:ce:b6:38:6f:94:df:0d:1c:61:41:6c:48:82:23:e6:6b:86:01:22:3a:f7:9a
:a4:60:5e:b2:5a:50:5d:40:ca:9a:9a:13:b1:8b:16:95:9a:b9:61:59:ff:e1:3b:7d:b9:e0:a
5:ea:36:ea:6b:21:8f:78:a9:d1:a5:9e:ee:ae:96:b9:96:62:53:ef:f5:1c:5e:fe:7f:aa:7a:
:68:95:6e:72:fc:ad:05:8d:29:e7:5f:55:26:af:7a:05:49:ed:8d:93:3b
```

Notes

For **show crypto key all**, the fields are:

Field	Description
Filename	Name of the file that contains the RSA keypair.
Bit Size	Size of the key pair in bits.
Type	Type of key exchange algorithm, such as RSA or DSA

For the **show crypto key <filename>** command, the modulus of the public key is also displayed.

Field	Description
Key Size	Size (in bits) of the key pair.
Modulus	Hex value of the public key modulus. The private key modulus is not shown for security purposes.

show crypto hardware

Provides information on the cryptographic acceleration hardware, if available.

Sample Output

```
ace19/Admin# show crypto hardware

=0x478860de, cpu_freq=0.6GHz, dwell=1, delta=600060015 (1s)
                   Total                          Delta
                   ------------------   --------------
obuf packets:      0x6f                            0        0.0 Packets/sec

Encrypt packets:   0x0                             0      nan Bytes/Packet
Decrypt packets:   0x0                             0      nan Bytes/Packet
Enc/Dec packets:   0x0                             0      nan Bytes/Packet
GP_OP packets:     0x6f                            0      nan Bytes/Packet
STX1 packets:      0x0                             0      nan Bytes/Packet
IMX1 packets:      0x0                             0      nan Bytes/Packet
IMX1 errors :      0x0                             0      nan Bytes/Packet
IMX1 drops  :      0x0                             0      nan Bytes/Packet

Encrypt bytes:     0x0                             0    0.000 Gbps
Decrypt bytes:     0x0                             0    0.000 Gbps
Enc/Dec bytes:     0x0                             0    0.000 Gbps
GP_OP   bytes:     0x5d820                         0    0.000 Gbps
STX1 bytes:        0x0                             0    0.000 Gbps
IMX1 bytes:        0x0                             0    0.000 Gbps

L3I Drop:          0x0                             0
L3I Fwd CP:        0x0                             0
L3I Fwd CP & DOS:  0x0                             0
L3I Decrypt Pass:  0x0                             0
L3I Total Pass:    0x0                             0

TX Backpressure:   0x0                             0  (STX1_BCKPRS_CNT)
RX Backpressure:   0x0                             0  (SPX1_BCKPRS_CNT)
TX Buffers used:   0x0                             0  (BMO_SP1_TPA)
TX Buffer:         0x0                                (High Water Mark)
RX Buffers used:   0x0                             0  (BMI_SP1_TPA)
RX Buffer:         0x0                                (High Water Mark)

enabled_cores:     0x3fffff
```

```
    This is a bit map of all enabled nitrox cores

available_cores:   0x3fffff
    This is a bit map of what nitrox cores are active at the time
    the stats were taken.

pom_robq_empty:    0x7fffffff  pom_inq_empty:      0x7fffffff
pom_tx0_outq_empty:0x0fffff    pom_tx1_outq_empty: 0x01ffff
POM count:         0(0)        Interrupts:        0x0(0)
```

Notes

The following counters are of interest when determining whether or not the Nitrox-II is "stuck":

Field	Description
STX1/IMX1 packets	Packets transmitted/received over SPI by the Nitrox-II. On a normal system these values should be equal when traffic has stopped flowing.
TX Buffers used	Buffers in use by the Nitrox-II for transmit
RX Buffers used	Buffers in use by the Nitrox-II for receive
available_cores	Shows which of the 22 Nitox-II cores are in use
POM count	Outstanding requests to the Packet Order Manager. The number outside the parentheses is the number of outstanding requests.
TX Backpressure	Number of SPI cycles that the Nitrox-II receives backpressure when trying to transmit data to the CDE.
RX Backpressure	Number of SPI cycles that the Nitrox-II exerts backpressure to the CDE.

Once traffic has stopped flowing, the RX/TX buffer counts and POM counts should go to 0. All cores should also be available, i.e., (Using:). If the values of the RX/TX buffers ever go above the value 0x800, then the chip has effectively crashed.

Also look at the TX Backpressure and RX Backpressure counters. High values for these indicate that the system is under some sort of stress.

show crypto session

Sample Output

```
ACE30001/Admin# show crypto session
SSL Session Cache Stats for Context
-----------------
Number of Client Sessions:                      0
Number of Server Sessions:                      0
```

Notes

This command displays the number of cached TLS and SSL client and server session entries in the current context. Therefore these sessions are eligible for session reuse.