Guide Contents
Main Article
Overview of ACE Troubleshooting
Understanding the ACE Module Architecture and Traffic Flow
Preliminary ACE Troubleshooting
Troubleshooting ACE Boot Issues
Troubleshooting with ACE Logging
Troubleshooting Connectivity
Troubleshooting ACE Appliance Ethernet Ports
Troubleshooting Remote Access
Troubleshooting Access Control Lists
Troubleshooting Network Address Translation
Troubleshooting ACE Health Monitoring
Troubleshooting Layer 4 Load Balancing
Troubleshooting Layer 7 Load Balancing
Troubleshooting Redundancy
Troubleshooting SSL
Troubleshooting Compression
Troubleshooting Performance Issues
ACE Resource Limits
Managing ACE Resources
Show Counter Reference

Contents

- 1 show np 1 me-stats -D
- 2 show np 1 me-stats -F
- 3 show np 1 me-stats -E
- 4 show np 1 me-stats -H
- 5 show np 1 me-stats -L global
- 6 show np 1 me-stats -L0
- 7 show np 1 me-stats -Q
- 8 show np 1 me-stats -b
- 9 show np 1 me-stats -c
- 10 show np 1 me-stats "-c c"
- 11 show np 1 me-stats -j
- 12 show np 1 me-stats -p
- 13 show np 1 me-stats -q queue
- 14 show np 1 me-stats -sappinspect
- 15 show np 1 me-stats -scde
- 16 show np 1 me-stats -scommon
- 17 show np 1 me-stats -sdrop
- 18 show np 1 me-stats -sfastpath
- 19 show np 1 me-stats -sfixup
- 20 show np 1 me-stats -shttp
- 21 show np 1 me-stats -sicm
- 22 show np 1 me-stats -sidle
- 23 show np 1 me-stats -slb
- 24 show np 1 me-stats -snitrox

- 25 show np 1 me-stats -snormalization
- 26 show np 1 me-stats -socm
- 27 show np 1 me-stats -sreass
- 28 show np 1 me-stats -sreceive
- 29 show np 1 me-stats -stcp
- 30 show np 1 me-stats -stimer
- 31 show np 1 me-stats -t
- 32 show np 1 me-stats -u
- 33 show np 1 me-stats -x
- 34 show np 1 me-stats -y
- 35 show np 1 memory
- 36 show np 1 nat policies
- 37 show np 1 reg

show np 1 me-stats -D

Sample Output

```
switch/RLB_test# show np 1 me-stats -D
iProxy: 0x69028fa7
oProxy: 0x2a01d2ec
Hash: 0x71
Real: 0x5
appHandle: 0x0x32c20280
iProxy: 0x69028fa7
oProxy: 0x2a01d2ec
Hash: 0x72
Real: 0x5
appHandle: 0x0x31df4f80
iProxy: 0x69028fa7
oProxy: 0x2a01d2ec
Hash: 0x73
Real: 0x5
2 entries open.
```

Notes

The command **show np [1|2] me-stats -D** displays the active proxy mapper data structures. These data structures are used to associate an inbound connection with an outbound connection. RLB uses these data structures because many requests can be received on a single inbound connection, and are transmitted on one of many outbound connections - so these structures keep track of connection association.

Therefore:

- iProxy is a pointer to the incoming connection
- oProxy is a pointer to the outgoing connection
- Hash is the hash index of this entry in the pmap hash table
- Real is the index of the rserver. Compare to the Rserver-id in the output for 'show cfgmgr internal table rserver'
- appHandle is a pointer to an application specific structure (stores user-name and/or csid if those types of sticky are also performed)

A pmap structure is created when a request is sent to the rserver, and it is removed when the response is received, or upon the configured inactivity timeout. Therefore this list shows the entries that are waiting for a response from the rserver. The NULL oProxy means that an outbound connection could not be created (because one already exists for that 5-tuple (source ip/port, dest ip/port, vlan), but it is in the opposite direction.

show np 1 me-stats -F

Displays the status of the internal SSL proxy structure associated with a vserver.

This command is invoked in this format:

```
show np 1 me-stats "-F<ctx_id> [v|V|0x<vserver_id>]"
```

Sample Output

```
switch/Admin# show np 1 me-stats "-F0 v"
context: 0   client vip_id: 0x00010000001 vip_flags: 0x00003
context: 0   client vip_id: 0x00010000002 vip_flags: 0x00001
context: 0   client vip_id: 0x00010000003 vip_flags: 0x00002
context: 0   client vip_id: 0x00010000004 vip_flags: 0x00003

switch/Admin# show np 1 me-stats "-F0 V"
context: 0   client vip_id: 0x00010000001 vip_flags: 0x00003
  config_ver: 1          sslproxy_id: 1
  cipher_list: 0x04
  rehandshake_data: 0 msgQdelay: 0          session_cache_timeout: 0
  n2_server_ctx:  f7ff80000000000 cert_size: 0    cert_req_msg_size: 0

context: 0   client vip_id: 0x00010000002 vip_flags: 0x00001
  config_ver: 1          sslproxy_id: 2
  cipher_list: 0x04
  rehandshake_data: 0 msgQdelay: 0          session_cache_timeout: 0
  n2_server_ctx:  f7ff00000000000 cert_size: 0    cert_req_msg_size: 0

switch/Admin# show np 1 me-stats "-F0 0x5b"
context: 0   server vip_id: 0x0000000005b vip_flags: 0x00003
  config_ver: 1          sslproxy_id: 47
  cipher_list: 0x04 0x05 0x09 0x0a 0x2f 0x35 0x03 0x60 0x08 0x62 0x64
  rehandshake_data: 0 msgQdelay: 0          session_cache_timeout: 0
  n2_server_ctx:  f7eb80000000000 cert_size: 1080 cert_req_msg_size: 0
  key_mod_ptr: 267520 key_mod_size: 256 key_exp_ptr: 268b48 key_exp_size: 3
```

show np 1 me-stats -E

Displays the crypto-related statistics for a single NP. The output is identical to **show stats crypto client/server**. See "show stats crypto server" for more information.

Sample Output

```
switch/Admin# show np 1 me-stats -E0
SSL Statistics:
------------------
SSL alert CLOSE_NOTIFY rcvd:                    0
SSL alert UNEXPECTED_MSG rcvd:                  0
```

```
SSL alert BAD_RECORD_MAC rcvd:                  0
SSL alert DECRYPTION_FAILED rcvd:               0
SSL alert RECORD_OVERFLOW rcvd:                 0
SSL alert DECOMPRESSION_FAILED rcvd:            0
SSL alert HANDSHAKE_FAILED rcvd:                0 The peer detected ssl handshake problems.
SSL alert NO_CERTIFICATE rcvd:                  0
SSL alert BAD_CERTIFICATE rcvd:                 0
SSL alert UNSUPPORTED_CERTIFICATE rcvd:         0
SSL alert CERTIFICATE_REVOKED rcvd:             0
SSL alert CERTIFICATE_EXPIRED rcvd:             0
SSL alert CERTIFICATE_UNKNOWN rcvd:             0
SSL alert ILLEGAL_PARAMETER rcvd:               0
SSL alert UNKNOWN_CA rcvd:                      0
SSL alert ACCESS_DENIED rcvd:                   0
SSL alert DECODE_ERROR rcvd:                    0
SSL alert DECRYPT_ERROR rcvd:                   0
SSL alert EXPORT_RESTRICTION rcvd:              0
SSL alert PROTOCOL_VERSION rcvd:                0
SSL alert INSUFFICIENT_SECURITY rcvd:           0
SSL alert INTERNAL_ERROR rcvd:                  0 The peer had internal problems.
SSL alert USER_CANCELED rcvd:                   0
SSL alert NO_RENEGOTIATION rcvd:                0
SSL alert CLOSE_NOTIFY sent:                    0
SSL alert UNEXPECTED_MSG sent:                  0
SSL alert BAD_RECORD_MAC sent:                  0
SSL alert DECRYPTION_FAILED sent:               0
SSL alert RECORD_OVERFLOW sent:                 0
SSL alert DECOMPRESSION_FAILED sent:            0
SSL alert HANDSHAKE_FAILED sent:                0
SSL alert NO_CERTIFICATE sent:                  0
SSL alert BAD_CERTIFICATE sent:                 0
SSL alert UNSUPPORTED_CERTIFICATE sent:         0
SSL alert CERTIFICATE_REVOKED sent:             0
SSL alert CERTIFICATE_EXPIRED sent:             0
SSL alert CERTIFICATE_UNKNOWN sent:             0
SSL alert ILLEGAL_PARAMETER sent:               0
SSL alert UNKNOWN_CA sent:                      0
SSL alert ACCESS_DENIED sent:                   0
SSL alert DECODE_ERROR sent:                    0
SSL alert DECRYPT_ERROR sent:                   0
SSL alert EXPORT_RESTRICTION sent:              0
SSL alert PROTOCOL_VERSION sent:                0
SSL alert INSUFFICIENT_SECURITY sent:           0
SSL alert INTERNAL_ERROR sent:                  0
SSL alert USER_CANCELED sent:                   0
SSL alert NO_RENEGOTIATION sent:                0
SSLv2 client hello received:                    0
SSLv3 client hello received:                    0
TLSv1 client hello received:                    0
SSLv3 negotiated protocol:                      0
TLSv1 negotiated protocol:                      0
SSLv3 full handshakes:                          0
SSLv3 resumed handshakes:                       0
Cipher sslv3_rsa_rc4_128_md5:                   0
Cipher sslv3_rsa_rc4_128_sha:                   0
Cipher sslv3_rsa_des_cbc_sha:                   0
Cipher sslv3_rsa_3des_ede_cbc_sha:              0
Cipher sslv3_rsa_exp_rc4_40_md5:                0
Cipher sslv3_rsa_exp_des40_cbc_sha:             0
Cipher sslv3_rsa_exp1024_rc4_56_md5:            0
Cipher sslv3_rsa_exp1024_des_cbc_sha:           0
Cipher sslv3_rsa_exp1024_rc4_56_sha:            0
Cipher sslv3_rsa_aes_128_cbc_sha:               0
Cipher sslv3_rsa_aes_256_cbc_sha:               0
```

```
TLSv1 full handshakes:                          0
TLSv1 resumed handshakes:                       0
Cipher tlsv1_rsa_rc4_128_md5:                   0
Cipher tlsv1_rsa_rc4_128_sha:                   0
Cipher tlsv1_rsa_des_cbc_sha:                   0
Cipher tlsv1_rsa_3des_ede_cbc_sha:              0
Cipher tlsv1_rsa_exp_rc4_40_md5:                0
Cipher tlsv1_rsa_exp_des40_cbc_sha:             0
Cipher tlsv1_rsa_exp1024_rc4_56_md5:            0
Cipher tlsv1_rsa_exp1024_des_cbc_sha:           0
Cipher tlsv1_rsa_exp1024_rc4_56_sha:            0
Cipher tlsv1_rsa_aes_128_cbc_sha:               0
Cipher tlsv1_rsa_aes_256_cbc_sha:               0
Total SSL client authentications:               0
Failed SSL client authentications:              0
SSL client authentication cache hits:           0
SSL static CRL lookups:                         0
SSL best effort CRL lookups:                    0
SSL CRL lookup cache hits:                      0
SSL revoked certificates:                       0
SSL CRL download failed:                        0
Total SSL server authentications:               0
Failed SSL server authentications:              0
Internal error:                                 0 The ssl-xscale had internal problems, possibly
                                                  due to "External buffer allocs failed".

Handshake FlushRX operations:                   0
Handshake FlushTX operations:                   0
Xscale messages rcvd from ME:                   0
Xscale messages sent to ME:                     0
Finish msg split across ssl recs:               0
Fasttx msg ring full:                           0
SSL_ME tx msg ring full:                        0
N2 encrypt_record:                              0
N2 decrypt_record:                              0
N2 random:                                      0
N2 handshake_hash:                              0
N2 hash:                                        0
N2 gpop_master:                                 0
N2 gpop_import_master_secret:                   0
N2 gpop_pkcs1v15enc:                            0
N2 gpop_pkcs1v15enc_crt:                        0
N2 gpop_finish:                                 0
N2 gpop_verify:                                 0
N2 gpop_pkcs1v15dec:                            0
N2 gpop_pkcs1v15dec_crt:                        0
N2 rsa_server_full:                             0
N2 resume:                                      0
```

show np 1 me-stats -H

Shows details on HTTP session entries. An HTTP session entry has a one-to-one correspondence to the proxy entry (-p) and the TCB entry (-t).

Syntax

Options:

```
-H [] [proxy_id|0xproxy_id|c|all]
-H: print first 10 HTTP session entries or all session entries if overall less than 10
-H proxy_id: print the HTTP session entry based on decimal proxy_id
-H 0xproxy_id:  print the HTTP session entry based on Hexadecimal proxy_id
-H c : print total number of open HTTP session entries
-H all: print all open HTTP session entries
```

Examples:

```
show np 1 me-stats -H
show np 1 me-stats -H 10
show np 1 me-stats -H 0xa
show np 1 me-stats -H c
show np 1 me-stats -H all
```

Sample Output

```
switch/Admin# show np 1 me-stats -H
        Debugging HTTP Connections
HTTP session for proxy entry: 0x1[seq=8]
------------------------------
Flags: 0x81080030                                       VS/Inspect ID: 0x1[seq=7]
        VS/Inspect ID: the LB vserver id or inspect policy id

Particles: First = 0xc0173f00                     Last = 0x80148060
        Nonzero "Particles" indicate buffered untransmitted data.
                The actual data can be shown using: show proc proc 1 me-stats "-P c0173f00"

Pipelined request particle: 0x0
        Pipelined request particle: is nonzero if the next pipelined request has not yet been pars

Parsed bytes: 26                                         Content-Length: 0
        Parsed bytes: number of HTTP bytes parsed
        Content-Length: header value, also used during url and chunk processing
        Result type:  enumerated value for the next or last parse result sent to LB/Inspect

Data state: 24                  Header type: 0                  Result type: 3
        Data state: the HTTP state machine status
        Header type: enumerated value for the last known header or method that was parsed

Regex state: 0x38000014 0x00000000
Hash State: 0x0                 Hash Bytes: 0x0                 Hash Index: 0
        Hash state:  64-bit hash accumulated hash state.  This is the value to be sent to LB/Inspe
        Hash bytes:  cache last 8-bytes of data to hash, since hash data may span packets
        Hash index:  index into Hash bytes.  We update Hash state when this reaches 8.

Temp offset: 5                  Insert length: 0
        Temp index:  used for temporary offset parsing, in order to e.g. store proper offsets
            for header value and other fields.
        Insert length: the number of bytes inserted

Scratch32: 0x11                 Scratch8: 0x0
        Scratch32, Scratch8:  temporary variables used during url deobfuscation, chunk decoding,
          header name parsing, etc.



HTTP session for proxy entry: 0x3[seq=0]
------------------------------
Flags: 0x817804e7                                       VS/Inspect ID: 0x1[seq=7]
Particles: First = 0x0                  Last = 0x0
Response status code: 200
Parsed bytes: 240                                       Content-Length: 893
```

```
Data state: 46                    Header type: 0                   Result type: 5
Regex state: 0x00000000 0x00000000
Hash State: 0x0                  Hash Bytes: 0x0                 Hash Index: 0
Temp offset: 0                    Insert length: 0
Scratch32: 0x0                    Scratch8: 0x0

2 HTTP proxy entries open.
```

show np 1 me-stats -L global

The "global" display shows the resources that are still globally available?that is, not allocated. You can use this command to track what resources are available on your system. (The output of "show resource usage all" provides the same output in a more readable format.)

Note that each connection takes two records within the system, so although you have say four million connections on the ACE, eight million connection records are created. Each IXP gets four million connection records?which is two million connections.

Sample Output

```
ace4/Admin# show np 1 me-stats -Lglobal
Resource limts for context :256
Rate                  Configured     Counters
Policer Name          Min      Max   min-toks max-toks   peak-toks    deny
       bandwidth:        0  8f0d180         0  8f0d0d6   5dccbc         0
 connection rate:        0    493e0         0    493df     5861         0
ssl-connections rate:    0      12c         0      12c        2         0
mgmt-traffic rate:       0  23c3460         0  23c3412   36528c         0
   mac-miss rate:        0      258         0      258      258         0
inspect-conn rate:       0      708         0      708        2         0

Resource              Configured        Counters
Policer Name          Min      Max      Min      Max     peak     deny
conc-connections:        0   249f00        0        5        0        0
mgmt-connections:        0      5dc        0        2        0        0
proxy-connections:       0    4cccc        0        0        0        0
ip-reassemble buffer:    0        0        0        0        0        0
  tcp-ooo buffer:        0        0        0        0        0        0
          regexp:        0        0        0        0        0        0
          xlates:        0    4cccc        0        0        0        0
acc-connections:         0       32        0        0        0        0
```

Notes

Field	Description
bandwidth	Bandwidth in bytes per second
connections	Connections per second
inspect-conn	rtsp/ftp inspect connections per second
mac-miss	MAC miss traffic punted to CP packets per second
mgmt-traffic	Management traffic bytes per second
ssl-connections	SSL connection rate
syslog	syslog messages per second

show np 1 me-stats -L0

Provides the same information as the regular non-debug **show resource usage [all]** command. Refer to the ACE documentation for a complete description: Command Reference.

Sample Output

```
ace4/Admin# show np 1 me-stats -L0
Resource limts for context :  0
Rate                   Configured     Counters
Policer Name           Min      Max   min-toks max-toks   peak-toks    deny
       bandwidth:        0  ee6b280        0  ee6ad54   5dc221        0
 connection rate:        0    7a120        0    7a11f     5859        0
ssl-connections rate:    0      1f4        0      1f4        2        0
mgmt-traffic rate:       0  3b9aca0        0  3b9aa60   3650b8        0
   mac-miss rate:        0      3e8        0      3e8      261   ec3570
inspect-conn rate:       0      bb8        0      bb8        2        0

Resource               Configured        Counters
Policer Name           Min      Max      Min      Max     peak     deny
conc-connections:        0   3d0900        0        5        c        0
mgmt-connections:        0      9c4        0        2        6        0
proxy-connections:       0    7ffff        0        0        5        0
ip-reassemble buffer:    0        0        0        0        0        0
  tcp-ooo buffer:        0        0        0        0        0        0
          regexp:        0        0        0        0        0        0
          xlates:        0    7ffff        0        0        0        0
```

show np 1 me-stats -Q

Displays message queue status information. Used to check the overall health status of the queues. A queue that is no longer working is indicated by a status of STUCK.

Sample Output

```
ACE30001/Admin# show np 1 me-stats -Q
Queue summary:
        lbrx 28712 28712 0 EMPTY
      lbrxhi 17488 17488 0 EMPTY
      lbtome  4082  4082 0 EMPTY
      sslxrx 15262 15262 0 EMPTY
    sslxtome  2336  2336 0 EMPTY
      ihmerx     0     0 0 EMPTY
      fasttx     7     7 0 EMPTY
          fp   232   232 0 EMPTY
        fphi     9     9 0 EMPTY
        airx     0     0 0 EMPTY
         icm   530   530 0 EMPTY
      slowtx  1598  1598 0 EMPTY
       reass  3562  3562 0 EMPTY
       ocmlo  2318  2318 0 EMPTY
       ocmhi     0     0 0 EMPTY
       tcprx  3374  3374 0 EMPTY
       tcptx   730   730 0 EMPTY
      httprx  3216  3216 0 EMPTY
      httptx  1608  1608 0 EMPTY
     fixuprx     0     0 0 EMPTY
```

```
     fixuptx     1     1 0 EMPTY
     sslmerx   192   192 0 EMPTY
   sslmerxhi  3944  3944 0 EMPTY
     sslmetx  3408  3408 0 EMPTY
     cmclose  2501  2501 0 EMPTY
      ipcplo     0     0 0 EMPTY
      ipcphi   226   226 0 EMPTY
     xtomelo  1863  1863 0 EMPTY
     xtomehi  3181  3181 0 EMPTY
       haxrx  3325  3325 0 EMPTY
        aitx  1069  1069 0 EMPTY
      syslog     0     0 0 EMPTY
       tcphp 28712 28712 0 EMPTY
    fasttxhi 17492 17492 0 EMPTY
```

Notes

The columns are:

- Queue name
- Head
- Tail
- Number of entries in queue
- Status of queue (STUCK, EMPTY, etc)

show np 1 me-stats -b

On the module, this command displays the particle buffers associated with the Fastpath and Slowpath on the IXP or dataplane (DP). The buffers are held per-Micro Engine (ME).

This command is issued when a IXP produces a ME CORE file and is most useful with a core file to see what data the DP was processing when the system crashed. On a live system it could be useful if a buffer leak was suspect and the ACE Escalation wanted to see if there was a pattern to the data in the buffers.

Sample Output

```
Fastpath and FastTX thread buffers
==================================

ME:1 thread:0 addr:0x0000 particle:0x40266be0 len:8397 rx_seq=8
0008  0x003820cd  0x0010cd27  0x00400100  0x0cccccd    .8 . ...' .@.. ....
000c  0x00059a3b  0x94d00032  0xaaaa0300  0x000c010b    ...; ...2 .... ....
0010  0x00000202  0x3c800000  0x0a8be304  0x0d000000    .... <... .... ....
0014  0x00800000  0x0a8be304  0x0d810100  0x00140002    .... .... .... ....
0018  0x000f0000  0x00000002  0x00cd0000  0x00000000    .... .... .... ....
001c  0x01030300  0x00000000  0x47454645  0x46434143    .... .... GEFE FCAC
0020  0x41000020  0x00010000  0xfafbfcfd  0xfeff0001    A..  .... .... ....
0024  0x02030405  0x06070809  0x44454346  0x43455046    .... .... DECF CEPF

ME:2 thread:0 addr:0x0000 particle:0xc0204670 len:78 rx_seq=0
0008  0x0500004e  0x00608034  0x00100028  0x00102880    ...N .`.4 ...( ..(.
000c  0x0000ffff  0xffffffff  0x001b539c  0xb1550800    .... .... ..S. .U..
0010  0x4500002c  0x40b10000  0xff112ae5  0x14141402    E.., @... ..*. ....
0014  0x14141401  0xc350c352  0x00180000  0x00010080    .... .P.R .... ....
0018  0x00000008  0x00000000  0x00000064  0x00000000    .... .... ...d ....

ME:21 thread:0 addr:0x0000 particle:0x002e3b60 len:72 rx_seq=0
```

```
0008  0x04400048  0x00508034  0x00000001  0x0005dc00    .@.H .P.4 .... ....
000c  0x00040012  0x43dc9300  0x001243dc  0x93000800    .... C... ..C. ....
0010  0x45000028  0xdbbe4000  0x4006a111  0x7f010122    E..( ..@. @... ..."
0014  0x0afa32e3  0xa1af01bb  0x8e867f84  0x16550688    ..2. .... .... .U..
0018  0x501016d0  0x0cb20000  0x00000000  0x00000000    P... .... .... ....

ME:22 thread:0 addr:0x0000 particle:0xc0204790 len:72 rx_seq=2
.....

SlowTX and FastTXHI thread buffers
==================================

ME:1 thread:7 addr:0x0230 particle:0x401f1210 len:48
0238  0x10200030  0x40000000  0x00000000  0x00000000    . .0 @... .... ....
023c  0x00003000  0x001e0000  0x00000201  0x2d3f0000    ..0. .... .... -?..
0240  0x00000000  0x0020001b  0x78bcaf46  0xc0a802c0    .... . .. x..F ....

ME:2 thread:7 addr:0x0230 particle:0x00000000 len:0

ME:21 thread:7 addr:0x0230 particle:0x00000000 len:0

ME:22 thread:7 addr:0x0230 particle:0x00000000 len:0
```

Notes

The "ME:##" notation in the output indicates the Micro Engine (ME) represented by the output.

show np 1 me-stats -c

This command shows details of connection records. You can choose to show a specific connection record or show all connection records. Detailed information about a connection ID is indicated by "-c <conn_id>"; all connections are shown if "-c" is selected with no <conn_id>.

Syntax

Options:

```
-c [] [conn_id|0xconn_id|c|all]?
-c: print first 10 connection records or all
    connection records if overall less than 10
-c conn_id: print the connection record based on
           decimal conn_id
-c 0xconn_id:  print the connection record based on
               Hexadecimal conn_id
-c c : print total number of open connections
-c all: print all connection records
```

Example:

```
show np 1 me-stats ?-c?
show np 1 me-stats ?-c 10?
show np 1 me-stats ?-c 0xa?
show np 1 me-stats ?-c c?
show np 1 me-stats ?-c all?
```

Sample Output

```
ACE30001/Admin#  show np 1 me-stats "-c 10"
Connection ID:seq: 10[0xa].3
  Other ConnID    : 60[0x3c].11
  Proxy ConnID    : 0[0x0].0
  Next Q    : 16777230[0x100000e]

0.222.173.0:50002 -> 192.168.5.179:50000 [RX-NextHop: Drop] [TX-NextHop: TX]
  Flags:  PAT: No  DynNAT: No  Implicit PAT: No On_Reuse: No
  L3 Protocol     : IPv4               L4 Protocol    : 17
  Inbound Flag    : 0
  Interface Match : Yes
    Interface MatchID: 0
  EncapsID:ver    : 0:0          TCP ACK delta  : 0x0
  MSS             : 0            TOS Stamp       : 0
  Repeat mode     : No           ARP Lookup      : No
  TOS Stamp       : No           TCP Window Check: No
  ACE ID          : 343          NAT Policy ID       : 0
  Post NAT hop    : 0
  Packet Count    : 0            Byte Count          : 0
  TCP Information: (State = 0)
    Window size   : 0            Window scale    : 0
    FIN seen      : No           FIN/ACK seen    : No
    FIN/ACK exp   : No           Close initiator : No
    FIN/ACK expval: 0            Last seq        : 0
   timestamp_delta: 0            Last ack        : 18eb9c
    No Trigger    : 0            Trigger Status   : 0
    Timestamp : 2942850f
  TCP options negotiated:
    Sack:Allow          TS:Allow        Windowscale:  Allow
    Reserved: Allow     Exceed MSS: Allow       Window var: Allow

Raw Connection Entry
0000  0x0100000e  0x00dead00  0xc0a805b3  0x11310000
0010  0xc352c350  0x00000000  0x00000000  0x00000000
0020  0x4b00003c  0x00000000  0x0008004f  0x00000000
0030  0x00000000  0x00000000  0x00000000  0x00000000
0040  0x00000000  0x0018eb9c  0x2942850f  0x00000000
0050  0x00000157  0x00000000  0x2942850f  0x00000000
0060  0x00000000  0x00000000  0x00000000  0x00000000
```

show np 1 me-stats "-c c"

Shows a per-IXP connection count. Similar to **show conn count**. (The -c switch must be in quotes.)

Sample Output

```
ACE30001/Admin# show np 1 me-stats "-c c"
27 connections open.
```

show np 1 me-stats -j

Shows the status of an Application Inspection (AI) proxy structure. The structure can be shown for a particular connection (by connection ID) or for all (with the **all** option).

Sample Output

```
ACE30001/Admin#  show np 1 me-stats -j <conn_id>

AI session for ProxyID.seq: %d.%d[0x%x.0x%x]
------------------------------
Flags: 0x........
Side Is Initiator: [yes|no]     L4 Protocol: [TCP|UDP]       Pending Fixup Response:
Ready For Unproxy: [yes|no]     Pending Unproxy: [yes|no]    Reproxy Conn: [yes|no]
TCP Flags Rcvd: FIN: [yes|no]   RST: [yes|no]                NO_MORE_DATA: [yes|no]
Particles: First = 0x..         Current = 0x...

Protocol: ....
FTP Command received =
FTP Protocol flags = 0x......
FTP First line reply code = 0x.....
FTP Data port received (real) = 0x.....
FTP Data port received (translated) = 0x.....

Raw AI Proxy Entry
.....
```

show np 1 me-stats -p

Displays the proxy mapper entries. The purpose of this structure is to maintain links between the basic connection structure and the higher level (L7) application structures.

Use this command to show details of the connection manager proxy entry. A proxy mapper entry is used for multicast protocols and maps an inbound connection to an outbound connection. Radius load-balancing utilizes pmap entries.

Syntax

```
-p [] [proxy_id|0xproxy_id|c|all]
-p: print first 10 proxy entries or all proxy entries if overall less than 10
-p proxy_id: print the proxy entry based on decimal proxy_id
-p 0xproxy_id:  print the proxy entry based on Hexadecimal proxy_id
-p all: print all open proxy entries
```

Sample Output

```
 ProxyID.seq: 4.3[0x4.0x3]
    Connection ID:  93571.13[0x16d83.0xd]
    Other Proxy ID:  8761.6[0x2239.0x6]
    Path ID:        2 [2]
    Other Path ID:  1 [1]
    ACE ID:         197 [c5]
    VS ID:          12 [c]
    Context ID:     0 [0]
    Policy ID:      5 [5]
    Real ID:        26 [1a]
    ParticlePtr:    [0]
    Packet offset:  [0]
```

Notes

Field	Description

Connection ID	Connection identifier of the basic connection for this proxy entry. (Try **show np 1 me-stats "-c 93571"**).
Other Proxy ID	Partner (inbound vs. outbound) proxy ID.
Path ID	A description of the path that packets take: • 0=FP, • 1=FP+TCP+HTTP, • 2=FP+TCP+SSL_HTTP, • 3=FP+TCP+FIXUP, • 4=FP+TCP+SSL+FIXUP, • 5=FP+FIXUP, • 6=FP+CP, • 7=FP+TCP+APPINSP, • 8=FP+TCP+HTTP+INSPHTTP)
Other Path ID	A mirror of the partner proxy's Path ID.
ACE ID	The aclmerge ACE ID which describes the features for this connection.
VS ID	The virtual server index for this connection.
Context ID	The context for this connection.
Policy ID	The policy chosen for this connection, following the LB decision.
Real ID	The real server ID chosen for this connection, following the LB decision.

show np 1 me-stats -q queue

Shows queue data for various queues. This would normally be run only as instructed by Cisco TAC.

The queue data takes the following general format:

```
1768: 15800018 00000000 a9b0e870 2c000004  [SSL Handshake / Normal]
```

The first column is the Queue Entry ID. The next four 32-bit columns are the queue data. This information is presented in hexadecimal format; therefore in the above example 15 = 21. The first of the four columns represent the message type (also shown in the last colunm--SSL Handshake, in this case). Of the other three columns, two are message specific, and the other is the message location in SRAM.

Sample Output

```
ACE30002/rlb_ssg# show np 1 me-stats -qsslmetx
sslmetx ring [BaseAddr(SRAM): 0xc0030000]
Head: 1759  Tail: 1759  Count:    0
1749: 038c0000 00000000 80250e00 23000003  [Packet / Raw]
1750: 15800018 00000000 a93e5330 23000003  [SSL Handshake / Normal]
1751: 15800008 00000000 a9350860 23000003  [SSL Handshake / Normal]
1752: 15800000 00000000 a8aea060 23000003  [SSL Handshake / Normal]
1753: 15800018 00000000 a9b24c10 23000003  [SSL Handshake / Normal]
1754: 15800000 00000000 a8b64ed0 23000003  [SSL Handshake / Normal]
1755: 038c0000 00000000 80250a50 23000003  [Packet / Raw]
1756: 12830080 00000000 00000000 23000003  [SSL Ctrl / RX Handshake Done]
1757: 0b000001 00000000 001ef210 23000003  [Data / Normal]
1758: 0b000006 00000000 00000000 23000003  [Data / Normal]
1759: 0b000006 00000000 00000000 21000004  [Data / Normal]<---HEAD<---TAIL
1760: 038c0000 00000000 0028ffd0 2c000004  [Packet / Raw]
1761: 15800008 00000000 a9b7fa80 2c000004  [SSL Handshake / Normal]
```

```
1762: 038c0000 00000000 c027f720 2c000004  [Packet / Raw]
1763: 038c0000 00000000 c020e020 2c000004  [Packet / Raw]
1764: 038c0000 00000000 c027f800 2c000004  [Packet / Raw]
1765: 15800018 00000000 a8b00a50 2c000004  [SSL Handshake / Normal]
1766: 15800008 00000000 a9b7f910 2c000004  [SSL Handshake / Normal]
1767: 15800000 00000000 a8aea060 2c000004  [SSL Handshake / Normal]
1768: 15800018 00000000 a9b0e870 2c000004  [SSL Handshake / Normal]
```

Notes

The output shown is for the qsslmetx queue. Other queues can be inspected with the switches shown below.

- -qaitx
- -qcmclose
- -qcntl0
- -qcntl1
- -qcntl2
- -qcntl3
- -qdata0
- -qdata1
- -qdata2
- -qdata3
- -qext0
- -qext1
- -qext2
- -qext3
- -qfasttx
- -qfasttxhi
- -qfixuprx
- -qfixuptx
- -qfp
- -qhaxrx
- -qhttprx
- -qhttptx
- -qicm
- -qihmerx
- -qipcphi
- -qipcplo
- -qlbrx
- -qlbtome
- -qocmhi
- -qocmlo
- -qreass
- -qslowtx
- -qsslmerx
- -qsslmetx
- -qsslxrx
- -qsslxtome
- -qsysl0
- -qsysl1
- -qsysl2
- -qsysl3
- -qtcphp
- -qtcprx
- -qtcptx
- -qxtomehi
- -qxtomelo

show np 1 me-stats -sappinspect

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-sappinspect -v"
AppInspect Statistics: (Current)
-------------------
Misc Errors:                                    0            0
Unsupported message type in AI ring:            0            0
DROP: AI proxy state seq mismatch:              0            0
DROP: AI other proxy state seq mismatch:        0            0
DROP: Global proxy state seq mismatch:          0            0
```

```
(Context ALL Statistics)

DROP: Conn entry seq mismatch errors:            0             0
DROP: Other conn entry seq mismatch erro         0             0
DROP: ACE version mismatch errors:               0             0
CONN RST: Inspect cfg ver mismatch error         0             0
NAF requests (pinhole) sent:                     0             0
NAF requests (all) sent:                         0             0
TCP unproxy requests sent:                       0             0
NAF responses received:                          0             0
TCP unproxy responses received:                  0             0
TCP data msgs received:                          0             0
Conn closed indication msgs received:            0             0
Parse Results from HTTP ME                       0             0
CONN RST: Protocol inspection errors:            0             0
CONN RST: Invalid inspect protocol id:           0             0
CONN RST: NAF response errors:                   0             0
CONN RST: Buffer allocation failures:            0             0
CONN RST: Packet rewrite failures:               0             0
CONN RST: NAF message creation errors:           0             0
CONN RST: NAF message send errors:               0             0
CONN RST: Total connections reset:               0             0
DROP: Invalid AI proxy state (Data) erro         0             0
DROP: Invalid TCP flags errors:                  0             0
DROP: Total packets dropped:                     0             0
TCP unproxies canceled:                          0             0
Pinholes opened:                                 0             0
No fixup info in NAF response:                   0             0
CONN RST: No fixup packet in NAF respons         0             0
DROP: Invalid AI proxy state (NAF) error         0             0
No particle in TCP data message:                 0             0
Total TCP connections processed:                 0             0
Total UDP connections processed:                 0             0
FTP: Drop - Data port too low:                   0             0
FTP: Drop - PORT cmd with no addr:               0             0
FTP: Drop - PORT cmd with third party ad         0             0
Skinny: Total Packets received:                  0             0
Skinny: Sessions created:                        0             0
Skinny: Sessions deleted:                        0             0
Skinny: Malformed packets:                       0             0
Skinny: Drop - policy-map:                       0             0
Skinny: Drop - protocol error:                   0             0
Skinny: Drop - registration enforcement:         0             0
Skinny: Drop - packet too small:                 0             0
Skinny: Drop - packet too large:                 0             0
Skinny: Drop - message-id out of range:          0             0
Skinny: Drop - Session creation error:           0             0
SIP: Sessions Created:                           0             0
SIP: Sessions Deleted:                           0             0
SIP: Session  Cloned:                            0             0
SIP: Packets  Received:                          0             0
SIP: Packets  Transmitted:                       0             0
SIP: Memory Allocation Failure:                  0             0
SIP: Drop - Message Path:                        0             0
SIP: Drop - Request Method:                      0             0
SIP: Drop - IM Susbcribe:                        0             0
SIP: Drop - Third Party Registration:            0             0
SIP: Drop - URI Length:                          0             0
SIP: Drop - Calling Party:                       0             0
SIP: Drop - Called Party:                        0             0
SIP: Drop - Content Length:                      0             0
SIP: Drop - Content Type:                        0             0
SIP: Drop - Non SIP Traffic:                     0             0
SIP: Drop - Strict Header Validation:            0             0
```

```
SIP: Drop - State Checking:                    0              0
SIP: Drop - Max Fowards:                       0              0
SIP: Drop - MalFormed Packets:                 0              0
SIP: Drop - IM Disbaled:                       0              0
SIP: Drop - Misc:                              0              0
SIP: Drop - Pmap Lookup Failure:               0              0
H225: Sessions Created:                        0              0
H225: Sessions Deleted:                        0              0
H225: Memory Allocation Failure:               0              0
H225: ASN1 Decode Failure:                     0              0
H225: ASN1 Encode Failure:                     0              0
H225: Packets Dropped:                         0              0
H225: Packets Transmitted:                     0              0
H225: Packets Received:                        0              0
H245: Sessions Created:                        0              0
H245: Sessions Deleted:                        0              0
H245: Memory Allocation Failure:               0              0
H245: ASN1 Decode Failure:                     0              0
H245: ASN1 Encode Failure:                     0              0
H245: Packets Dropped:                         0              0
H245: Packets Transmitted:                     0              0
H245: Packets Received:                        0              0
RAS: Sessions Created:                         0              0
RAS: Sessions Deleted:                         0              0
RAS: Memory Allocation Failure:                0              0
RAS: ASN1 Decode Failure:                      0              0
RAS: ASN1 Encode Failure:                      0              0
RAS: Packets Dropped:                          0              0
RAS: Packets Transmitted:                      0              0
RAS: Packets Received:                         0              0
H323: Tunnel packets blocked:                  0              0
H323: Call Party Number:                       0              0
H323: H225 state machine check:                0              0
H323: RAS State machine check:                 0              0
H323: Call duration:                           0              0
FTP: Error - Free FTP state on outbound        0              0
```

Notes

Field	Description
Misc Errors	Connection ID is NULL while processing HTTP/NAF messages
Unsupported message type in AI ring	Unknown message type/msg_id is received from other modules
DROP: AI proxy state seq mismatch	AI proxy sequence number mismatch while reading proxy state
DROP: AI other proxy state seq mismatch	AI other side proxy sequence number mismatch while reading state
DROP: Global proxy state seq mismatch	AI global proxy sequence number mismatch while reading state
DROP: Conn entry seq mismatch errors	AI connection entry sequence number mismatch
DROP: Other conn entry seq mismatch erro	Other side connection entry sequence number mismatch

DROP: ACE version mismatch errors	ACL config outdated/action node dirty
CONN RST: Inspect cfg ver mismatch error	Inspect config outdated/mismatch
NAF requests (pinhole) sent	Nat Appl (FTP, RTSP, SIP) Fixup pinhole request sent
NAF requests (all) sent	Nat Appl Fixup total messages posted to OCM_LO_PRI_MSG_RING
TCP unproxy requests sent	TCP unproxy requests sent
NAF responses received	Total number of Nat Appl Fixup responses received
TCP unproxy responses received	Total number of TCP unproxy responses received
TCP data msgs received	Total number of data messages from TCP module
Conn closed indication msgs received	Connection closed message from CM
Parse Results from HTTP ME	Parse Result message from HTTP
CONN RST: Protocol inspection errors	Drop current packet and reset the connection on protocol errors
CONN RST: Invalid inspect protocol id	Invalid/unsupported inspect type/ID
CONN RST: NAF response errors	Nat Appl Fixup response errors
CONN RST: Buffer allocation failures	Buffer allocation for sending NAF request failed
CONN RST: Packet rewrite failures	buf_chain_replace failed while trying to rewrite the packet
CONN RST: NAF message creation errors	NAF request creation failure invalid nat_app_fixup_info or buf alloc
CONN RST: NAF message send errors	Not used
CONN RST: Total connections reset	Connection reset due to any reasons
DROP: Invalid AI proxy state (Data) erro	Not used
DROP: Invalid TCP flags errors	TCP flag is begin_data but proxy state exists
DROP: Total packets dropped	Particle chain dropped due to any reason
TCP unproxies canceled	TCP unproxies canceled by AppInspect
Pinholes opened	Total number of pinholes created successfully
No fixup info in NAF response	NULL fixup info in the NAF response
CONN RST: No fixup packet in NAF respons	No fixup particle in NAF response (not used)
	Invalid AI proxy state while processing NAF response

DROP: Invalid AI proxy state (NAF) error	
No particle in TCP data message	NULL particle in non-FIN, non-RST data msg
Total TCP connections processed	Total TCP connections that AI processed
Total UDP connections processed	Total UDP connections that AI processed
CONN RST: FTP data port too low	FTP data port too low; data port less than 1024
CONN RST: FTP PORT cmd with no addr	No address in FTP PORT command
CONN RST: FTP PORT cmd with third party	The IP address in the PORT command (active FTP) or the IP address in the response to the PASV command (passive FTP) is not the sender's IP address.
Skinny: Total Packets received	Total Packets received skinny inspect layer
Skinny: Sessions created	Total sessions created for skinny inspect
Skinny: Sessions deleted	Skinny sessions released after AI processing
Skinny: Malformed packets	Not used
Skinny: Drop - policy-map	Skinny session dropped due to policy map check failure
Skinny: Drop - protocol error	Skinny session dropped due to protocol error
Skinny: Drop - registration enforcement	Skinny session dropped due to registration not completed
Skinny: Drop - packet too small	Skinny session dropped due to packet too small
Skinny: Drop - packet too large	Skinny session dropped due to packet too large
Skinny: Drop - message-id out of range	Skinny session dropped due to message ID out-of-range
Skinny: Drop - Session creation error	Skinny session creation error due to out-of-memory
SIP: Sessions Created	Total SIP sessions created
SIP: Sessions Deleted	Total SIP sessions deleted
SIP: Session Cloned	Total no of SIP sessions cloned based on the flags
SIP: Packets Received	Total no of SIP packet received
SIP: Packets Transmitted	Total number of SIP packet transmitted
SIP: Memory Allocation Failure	Out-of-memory for packet buffer
SIP: Drop - Message Path	SIP filter drop due to message path
SIP: Drop - Request Method	SIP filter drop due to request method
SIP: Drop - IM Susbcribe	SIP filter drop due to IM subscriber

SIP: Drop - Third Party Registration	SIP filter drop due to third party registration
SIP: Drop - URI Length	SIP filter drop due to URI length
SIP: Drop - Calling Party	SIP filter drop due to calling party
SIP: Drop - Called Party	SIP filter drop due to called party
SIP: Drop - Content Length	SIP filter drop due to content length
SIP: Drop - Content Type	SIP filter drop due to content type
SIP: Drop - Non SIP Traffic	SIP filter drop due to non-SIP Traffic
SIP: Drop - Strict Header Validation	SIP filter drop due to strict header validation
SIP: Drop - State Checking	SIP filter drop due to state checking
SIP: Drop - Max Fowards	SIP filter drop due to Max Fowards
SIP: Drop - MalFormed Packets	SIP filter drop due to malformed Packets
SIP: Drop - IM Disbaled	SIP filter drop due to IM disabled
SIP: Drop - Misc	SIP filter drop due to misc errors like failed to create Timer
SIP: Drop - Pmap Lookup Failure	SIP filter drop due to Proxy mapper lookup failure
FTP : Error - Free FTP state on outbound	FTP state free on the wrong outbound side

show np 1 me-stats -scde

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-scde -v"
CDE Statistics: (Current)
--------------
Error drops:                                    0             0
Runts errors:                                   0             0
Invalid header errors:                          0             0
Protocol errors:                                0             0
Not parsed errors:                              0             0
Max length errors:                              0             0
Length errors:                                  0             0
Packet CRC errors:                              0             0
DBUS FCS errors:                                0             0
L3 checksum errors:                             0             0
L4 checksum errors:                             0             0
Bad CDE dst errors:                             0             0
Bad CDE length errors:                          0             0
```

Notes

Field	Description
Error drops	

	0 sum of l4_xsum_error + L3_xsum_error + fcs_error + crc_error + length_mismatch + max_len_exceeded + pkt_not_parsed + protocol_error
Runts errors	Deprecated
Invalid header errors	Deprecated
Protocol errors	Hyperion Bus Protocol Error Occurred. This flag is set when Hyperion Bus SOP/EOP packets are received at an unexpected time.
Not parsed errors	The CDE parser did NOT successfully determine the L2/L3/L4 types and these packets were not aligned and thus dropped.
Max length errors	Packet Length exceeded the maximum packet length, as specified by the HR_MAX_PACKET_LENGTH Register (or imph->max_len_exceeded set).
Length errors	Length specified in the DBus Header does not match the number of bytes received before encountering EOP.
Packet CRC errors	Ethernet CRC Error Detected.
DBUS FCS errors	DBus Header FCS Error Detected.
L3 checksum errors	IPv4 Header Checksum Error Detected.
L4 checksum errors	TCP/UDP Checksum Error Detected.
Bad CDE dst errors	Invalid IMPH destination or IMPH destination not matching on the IXP (fastpath_imph_sanity_check).
Bad CDE length errors	Deprecated

show np 1 me-stats -scommon

This command displays counters that are common to all MEs. If this output shows possible issues, the individual ME output can be examined to identify a problematic ME.

Sample Output

```
Module
======
ace-1/Admin# show np 1 me-stats -scommon
Common Statistics: (Current)
-------------------
Connection Lookup Abort Count:                    0
Connection Lookup Abort Last:                     0
Internal buffers allocated:                17840731
Internal buffers released:                 17840687
Internal buffer allocs failed:                    0
External buffers allocated:                      25
External buffers released:                       18
External buffer allocs failed:                    0
Syslog buffers allocated:                         0
Syslog buffers released:                          0
Syslog buffer allocs failed:                      0
Control buffers allocated:                        0
Control buffers released:                         0
Control buffer allocs failed:                     0
Hash lock contention count:                       0
X TO ME Pkt count:                          4941691
```

```
Internal buffers 0 ref cnt:                        0
External buffers 0 ref cnt:                        0
Debug stat 1:                                      0
Debug stat 2:                                      0
Debug stat 3:                                      0
Debug stat 4:                                      0
Debug stat 5:                                      0
Debug stat 6:                                      0
Debug stat 7:                                      0
Debug stat 8:                                      0
Debug stat 9:                                      0

Appliance
=========
scim-2/Admin# show np 1 me-stats -scommon
Common Statistics: (Current)
-------------------
Internal buffers allocated:              1370432141
Internal buffers released:               1370432129
External buffers allocated:                      15
External buffers released:                       15
Hash lock contention count:                       3
connection freestack lock contention:          2140
proxy freestack lock contention:                  0
Connection Lookup Abort Count:                    0
Connection Lookup Abort Last:                     0
Internal buffer allocs failed:                    0
External buffer allocs failed:                    0
Syslog buffer allocs failed:                      0
Control buffer allocs failed:                     0
Syslog buffers allocated:                         0
Syslog buffers released:                          0
Control buffers allocated:                        0
Control buffers released:                         0
X TO ME Pkt count:                                0
Internal buffers 0 ref cnt:                       0
External buffers 0 ref cnt:                       0
Buffer-share decrement failed:                    0
Pkts processed by core-0:                   9690617
Pkts processed by core-1:                 115703116
Pkts processed by core-2:                  95953088
Pkts processed by core-3:                  81256728
Pkts processed by core-4:                  69912542
Pkts processed by core-5:                  61227819
Pkts processed by core-6:                  52382013
Pkts processed by core-7:                  56820018
Pkts processed by core-8:                  46564394
Pkts processed by core-9:                  40868861
Pkts processed by core-10:                 38325180
Pkts processed by core-11:                 37564215
Pkts processed by core-12:                 35873171
Pkts processed by core-13:                 41323348
Pkts processed by core-14:                 41070922
Pkts processed by core-15:                 34651040
```

Notes

Field	Description
Internal buffer-related counters	Internal buffers hold data in a common place so all MEs can access the data. External buffers hold pointers to the data in internal buffers. In general, buffers released should be within a reasonable value of buffers allocated. Too large a difference could indicate buffer

	release problems or buffer leaks. There are 256k Internal buffers and 64k external buffers. If buffer allocation exceeds 75%, no more buffers are allocated and ACE will stop accepting new connections.
Syslog buffers allocated	Buffers allocated for configured SYSLOG hosts.
Syslog buffers released	Buffers freed for configured SYSLOG hosts.
Syslog buffer allocs failed	The number of time the ACE could not get a buffer for SYSLOG and had to drop the message. If there is a large difference between Syslog buffers allocated/released this could indicate a SYSLOG buffer leak.
Control buffers allocated	Buffers allocated for configured ACE Internal CONTROL messaging.
Control buffers released	Buffers freed for configured ACE Internal CONTROL messaging.
Control buffer allocs failed	The number of time the ACE could not get a buffer for internal control messages. If there is a large difference between CONTROL buffers allocated/released this could indicate a CONTROL buffer leak.
Hash lock contention count	FP hashes the SrcIP, SrcPort, DstIP, DstPort to find an entry (called a bin, 4 bytes in size) in the connection hash table. There are 2 million bins in each IXP that live in SRAM. If the bin identified is locked (another thread has already acquired bin), the process blocks until the bin becomes available and this counter is incremented.
X TO ME Pkt count	These are packets that Xscale has sent to ME - SSL, App inspect and heartbeat.
Internal buffers 0 ref cnt	Incremented when Xscale tries to release a buffer when there are no buffers to be released.
External buffers 0 ref cnt	Non-zero number indicates buffers released multiple times. These are for DE use in engineering images.
Debug stat 1 through 9	Not used.

show np 1 me-stats -sdrop

Sample Output

```
DAR1_ACE_Piter/Admin# show np 1 me-stats "-sdrop -v"
Fastpath Statistics: (Current)
-------------------
DROP: RX Interface miss:                    234395
DROP: Unknown Msg received:                2258878
DROP: Bandwidth rate policed:                    7
(Context ALL Statistics)
(No relevant stats)

Receive Statistics: (Current)
------------------
(No relevant stats)

ICM Statistics (Current)
--------------
If lookup error:                              10
(Context ALL Statistics)
```

```
Route lookup Error:                                5

OCM Statistics: (Current)
--------------
(No relevant stats)
(Context ALL Statistics)
(No relevant stats)

HTTP Statistics (Current)
--------------
(No relevant stats)
(Context ALL Statistics)
(No relevant stats)

FIXUP DNS  Statistics (Current)
--------------
(No relevant stats)
(Context ALL Statistics)
(No relevant stats)

CDE Statistics: (Current)
--------------
(No relevant stats)

Crypto Statistics: (Current)
------------------
(No relevant stats)

Nitrox Error Statistics: (Current)
------------------
(No relevant stats)

REASSEMBLY Statistics: (Current)
--------------
(No relevant stats)

AppInspect Statistics: (Current)
-------------------
(No relevant stats)

(Context ALL Statistics)

(No relevant stats)

DAR1_ACE_Piter/Admin#
```

Notes

Field	Description
Fastpath Statistics	Summary of drop counters from "show np 1 me-stats -sfp"
Receive Statistics	Summary of drop counters from "show np 1 me-stats -srx"
ICM Statistics	Summary of drop counters from "show np 1 me-stats -sicm"
OCM Statistics	Summary of drop counters from "show np 1 me-stats -socm"
HTTP Statistics	Summary of drop counters from "show np 1 me-stats -shttp"
FIXUP DNS Statistics	Summary of drop counters from "show np 1 me-stats -sfixup"
CDE Statistics	Summary of drop counters from "show np 1 me-stats -scde"
Crypto Statistics	Summary of drop counters from "show np 1 me-stats -scrypto"

Nitrox Error Statistics	Summary of drop counters from "show np 1 me-stats -snitrox"
REASSEMBLY Statistics	Summary of drop counters from "show np 1 me-stats -sreass"
AppInspect Statistics	Summary of drop counters from "show np 1 me-stats -sappinspect"

show np 1 me-stats -sfastpath

This command is identical to the **show np 1 me-stats -sfp** command.

Sample Output

```
ACE30002/Admin# show np 1 me-stats -sfastpath
Fastpath Statistics: (Current)
-------------------
Errors:                                      610             0
FPTX Hi Priority receive:                  13473652          26
Fastpath pkt received:                     26294092          49
FPTX receive:                               6673160          13
FastTX receive:                             6015961          10
SlowTX receive:                              131350           0
Packets transmit to hyperion:               7169543          10
Packets punt to CP:                         1146277           1
Packets punt to Nitrox:                      131331           0
Packets punt to other IXP:                     1682           0
Packets forward to ICM:                     6054324          10
Packets IPCP forward:                            72           0
Large buffer TX count:                       27835           0
DROP: Connection Miss:                         288           0
DROP: RX Interface miss:                     973448           2
DROP: Unknown Msg received:                 4677427          16
DROP: Bandwidth rate policed:                    7           0
Close request Sent:                          14554           0
Packets forward to SSL-XScale:               131331           0
Wait for empty TFIFO:                        106949           0
Drop: Virtual MAC packets to standby:         24138           0
(Context ALL Statistics)
Packets forward to Reassembly:                   96           0
Packets forward to XScale:                  5980682          10
DROP: Connection Route:                        2863           0
Drop: Invalid connection hit:                   134           0
```

Notes

Field	Description
Errors	This counter in the fastpath is usually incremented when there is a normalization failure. The output of **sh np <x> me-stats "-s normalization"** will list the failure reasons.
FPTX Hi Priority receive	Count of packets received from the network with a high Class of Service (COS) value, or from another chip as high priority. Generally HA traffic from the network and SSL traffic from Nitrox will be classified as high priority.
Fastpath pkt received	Count of packets received by fastpath from all sources, including fptx, fptx_hi, fasttx, and slowtx.
FPTX receive	Count of all packets received from the Receive Micro Engine (ME).
FastTX receive	Count of packets received by fastpath, sent by an ME other than Receive, and destined for somewhere other than the Broadcom chip. Examples include packets from the reassembly

	ME, initial packets from connection manager, Layer 7 traffic forwarded by TCP, and SSL traffic destined for the Nitrox (SSL) chip.
SlowTX receive	Count of packets received by fastpath from an ME other than Receive, and destined for the Broadcom chip. Mostly Internet Protocol Control Protocol (IPCP) traffic.
Packets transmit to hyperion	Count of packets transmitted by fastpath, destined for the network. Hyperion is the interface chip between the ACE and the Cat6k (SUP) backplane.
Packets punt to CP	Count of packets forwarded to the Broadcom Control Plane (CP) chip.
Packets punt to Nitrox	Count of packets forwarded to the Nitrox SSL chip.
Packets punt to Daughtercard	Count of packets sent to one of the two daughterboards.
Packets punt to other IXP	Count of packets sent from this IXP to the other IXP.
Packets transmitted (loopback)	Number of IMPH loopback packets transmitted.
Debug packet copy to CP	Deprecated statistic (used for debug).
Packets forward to ICM	Count of packets forwarded to the Input Connection Manager (ICM) ME. Includes connection misses, and recognized bridge traffic.
Packets forward to OCM	Count of packets forwarded to the Output Connection Manager (OCM) ME.
Packets forward to TCP	Count of packets forwarded to the TCP ME. Mostly L7 traffic.
Packets forward to Fragmentation	Count of packets forwarded to the Fragmentation and Reassembly ME for fragmentation.
Packets IPCP forward	Count of IPCP packets recognized and forwarded to another ME.
Large buffer TX count	Count of packets which span more Transmit First-in-First-Out (TFIFO) elements than we have in our burst window (2k bytes). When backpressure is encountered by one of these packets, the buffer must be aborted and sent, rather than skipped.
WARN: TX Packet too small	Indicates the pre-calculated TFIFO count for the buffer in question was larger than needed. Non-fatal condition, but usually indicates a logic or hardware error, or memory corruption.
DROP: Packet too big error	Indicates the pre-calculated TFIFO count for the buffer in question was smaller than needed. Packet will be dropped. Usually indicates a logic or hardware error, ormemory corruption.
DROP: Connection Miss	Drop of a packet not associated with an existing connection, when connection miss rate has been exceeded.
DROP: Bad connection route	Drop of a packet where the internal "next hop" is invalid. This stat indicates a late drop by fastpath. Reasons include: • TCP normalization error (most drops are from this, and are due to TCP data past FIN) • Packets received after a connection has been closed (via timeout, normal shutdown, or error condition)

	• Packets received while in standby stat
DROP: RX Interface miss	Could not find an associated interface ID. Packets are received which either are not on a configured VLAN, do not match our MAC (either shared or burnt-in), or match an interface which has been invalidated. This may indicate a configuration issue.
DROP: Out of buffers	fastpath_lmem_to_dram failure
DROP: Unknown Msg received	This usually means that a packet from the network was unclassified by the ACE fastpath. This is normal and can generally be ignored. The typical cause is non-IP traffic. If all packets received are being dropped as unknown, then there is likely a problem.
DROP: Bandwidth rate policed	This is the number of packets dropped as a results of hitting the licensed bandwidth_rate or connection_rate limit.
Close request Sent	Fastpath generated a close message and sent it to cm_close. This is normal to be incremented.
Packets dropped (encap invalid)	This statistic is incremented when the Layer 2 (L2) adjacency info (aka encaps entry) has been marked as invalid.
Close request Sent: (encap mismatch)	This statistic is incremented when the encaps entry version number does not match the version in the connection record. The connection is closed as well.
Packets forward to SSL-ME	Count of packets forwarded to the SSL ME.
Packets forward to SSL-XScale	Count of packets forwarded directly to the SSL code running on the XScale processor.
Ack trigger msgs sent	This statistic is incremented when the a Send Ack Trigger Message is sent due to the packet ack number being greater than the stored ack number (send ack trigger when packet ack > stored ack).
DROP: TO CP rate policed	This statistic is incremented when there is a "to_cp_rate_police" failure.
Wait for empty TFIFO	This counter is incremented via the command "fastpath_increment_stat(fastpath_transmit_wait)" in tx_validate_window_check(). This counter increments when the Fastpath ME is waiting for an available buffer. This counter increments once roughly every 6 nanoseconds while waiting (it is a busy wait loop). Fastpath will wait forever until a buffer is ready, so if a buffer is never available, the FP ME would 'crash' (i.e., loop forever and appear to be stuck, resulting in an ME core). This counter seems to increment in otherwise normal configurations.
FastQ Transmit Backpressure	A count of times that an attempt to transmit a packet on the FastQ channel was refused due to backpressure from the Classification and Distribution Engine (CDE). This usually results in a drop.
SlowQ Transmit Backpressure	SlowQ is used for to-CP traffic, including ARP, probes, remote sessions, etc. This counter indicates that this to-CP traffic is being dropped, possibly because the CP is overwhelmed.
Hyperion Transmit Backpressure	Hyperion Backpressure is caused when Cat6k's (SUP) input queues are full. Backpressure and Backpressure drops can be seen when SPAN is over subscribed. The same condition can occur if Cat6k's is using older linecards.
Drop: Transmit Backpressure	Count of packet drops due to backpressure.

Drop: Virtual MAC packets to standby	Drop all packets except to/from CP and those required for high availability (HA) when the ACE is standby.
Drop: Shared MAC in non-shared interface	Count of packets dropped when received with a shared MAC on an unshared interface.
Drop: Next-Hop queue full	The internal routines fastpath_xsmit_mem or fastpath_xmit_fragment failed to post a message to the other ME's ring queue.
Drop: Diag to SSL-ME	Not used.
Diag packets forwarded to SSL-ME	Not used.
Drop: Invalid IMPH Destination	The valid destinations are ixp0, ixp1, nitrox, hyperion, broadcom?called from function fastpath_xmit_packet. These are packets that are intercepted late in the transmit pipeline, and usually indicate a buffer corruption issue.
Drop: Invalid IMPH Next-Hop	Not expected to occur.
Drop: IP DF bit set	The packet is dropped because the Ipv4 Do Not Fragment (DF) bit was set in the IP Header.
Drop: No fragmentation of L3 Encap	fastpath_xmit_fragment - packet of type L3_ENCAP (Layer3 encapsulation of frame) are not supposed to be fragmented. PACKET_L3_ENCAP is used by TCP and UDP to send data out of ACE. Basically, TCP/UDP header is built (without populating ports) and given to the fastpath as L3_ENCAP, which tells fastpath to construct the Layer 3 header and also to populate the ports. Fragmentation is not required for TCP packets, as TCP is supposed to send a maximum of Maximum Segment Size (MSS) worth of data and hence the error statistic "Drop: No fragmentation of L3 Encap". This error stat is applicable only for TCP and not for UDP.
FastPath Jumbo pkt retransmit on BP	Count of the number of times that the fastpath attempts to retransmit a packet which is both high priority (meaning it can't be dropped) and large (meaning that it won't fit into the 2k burst window).
Drop: exceed buffer threshold limit	The packet is dropped for a proxied connection if buffers are heavily used - threshold is 88% of MINIMUM_PARTICULE_CNT and SRAM_PARTICULE_CNT.
The following are context ALL statistics	
Packets forward to Reassembly	Count of packets forwarded to Reassembly and Fragmentation ME for Reassembly.
Packets forward to XScale	Count of packets forwarded to the XScale ME.
DROP: Connection Route	Count of packets dropped because fastpath determines that the packet should not be forwarded to its internal destination for some reason.
Packets forward, reproxy	Count of packets forwarded to TCP for reproxy, because the reproxy condition was met.
Packets forward, reproxy	Count of packets forwarded to TCP for reproxy, because the reproxy trigger was met. The reproxy trigger is "packet sequence number" plus "payload length" is greater than "stored

w/trigger	sequence number".
Drop: Invalid connection hit	Count of packets dropped because they matched a connection which is not in a valid state. This usually happens when the connection is in the process of being released when a new packet on that connection is received.
Drop: Reproxy out of order	The TCP segment is out of order. Dropping these out-of-order packets is intended to cause retransmissions; it should not cause ACE to drop connections. Forcing the peer to retransmit should allow it to get back in order. However, the TCP packet that caused a reproxy need to be retransmitted, or a data error occurs.

show np 1 me-stats -sfixup

This command displays counters related to the fixup microengine. This microengine inspects traffic of various protocols to protect against various attacks and malformed messages.

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-sfixup -v"
FIXUP DNS  Statistics (Current)
--------------
Unknown rx msgs received:                        0             0
Unknown tx msgs received:                        0             0
DNS dest decision conn. over limit:              0             0
DNS conn control connection over limit:          0             0
DNS nat app fixup connection over limit:         0             0
DNS packet connection over limit:                0             0
DNS app NAT fixup error:                         0             0
Wrong context ID:                                0             0
App_id entries timed-out:                        0             0
Forward TFTP packets:                            0             0
DNS proxy entries timed-out:                     0             0
Close tx msgs received:                          0             0
Close msg connetion ID mismatch:                 0             0
(Context ALL Statistics)
Packet rx msgs received:                         0             0
Data rx msgs received:                           0             0
Nat app fixup tx msgs received:                  0             0
DNS invalid session:                             0             0
DNS query received:                              0             0
DNS query forwarded:                             0             0
DNS response received:                           0             0
DNS response forwarded:                          0             0
Hash entries inserted:                           0             0
Hash entries deleted:                            0             0
Hash entries updated:                            0             0
Hash miss errors:                                0             0
DNS offset error:                                0             0
DNS offset error2:                               0             0
ICMP packet received:                            0             0
ICMP error packet received:                      0             0
Fixup IPCP msgs sent:                            0             0
app_id allocation error:                         0             0
app_id empty error:                              0             0
No packet in chain:                              0             0
NAT app fixup response error:                    0             0
DNS loop error:                                  0             0
DNS label error:                                 0             0
DNS name error:                                  0             0
DNS response count error:                        0             0
```

```
DNS response wrong dnsdp:                          0              0
DNS response no buffer:                            0              0
DNS connection mismatch:                           0              0
DNS query parse error:                             0              0
DNS response parse error:                          0              0
DNS response rr parse error:                       0              0
DNS response packet length error:                  0              0
DNS query packet length error:                     0              0
DNS query l5 offset error:                         0              0
DNS query packet type error:                       0              0
DNS close error:                                   0              0
DNS NAT connection mismatch error:                 0              0
DNS cptr range error:                              0              0
DNS proxy conn is in use:                          0              0
Fixup Protocol ID error:                           0              0
Other error:                                       0              0
NAF Error (bad naf request):                       0              0
NAF Error (pinhole exists for different            0              0
NAF Error (pinhole exists and dup not al           0              0
NAF Error (tracker create):                        0              0
NAF Error (tracker link):                          0              0
NAF Error (total):                                 0              0
Pinhole creates:                                   0              0
Tracker links:                                     0              0
Fixup IPCP msgs recd:                              0              0
Drop: Fastpath Queue Full (DNS):                   0              0
```

Notes

Field	Description
Unknown rx msgs received	Unknown message ID type of message was received on CNBAR_RX_MSG_RING.
Unknown tx msgs received	Unknown message ID type of message was received on CNBAR_TX_MSG_RING.
DNS dest decision conn. over limit	Not used.
DNS conn control connection over limit	Resource usage exceeded by DNS proxy connection; over the proxy entry limit, 256k.
DNS nat app fixup connection over limit	When doing DNS NAT, the proxy ID in the app_fixup_info discovered to be over the proxy entry limit.
DNS packet connection over limit	Not currently used.
DNS app NAT fixup error	Will process naf response for DNS, app_fixup_info return error from nat_fixup or failed read proxy for this connection.
Wrong context ID	When increment context-specific counter, find the passed in context_id over limit of MAXIMUM_CONTEXT_QTY.
Forward TFTP packets	Fixup transmission queue forwarding tftp packet after pinehole creation.
Close tx msgs received	Number of close messages from cm_close received by fixup transmission queue.
Close msg connetion ID mismatch	While fixup transmission processed close message, the dns_proxy session became invalid or the sequence of the DNS proxy entry mismatched.
DNS invalid session	DNS proxy session invalid (seq mismatch or flag invalid).
DNS query received	Received DNS query.

DNS query forwarded	Successfully processed DNS query and forward it out.
DNS response received	Received DNS response.
DNS response forwarded	Successfully processed DNS response and forwarded it out.
Hash entries inserted	Fixup insert hash entry for the app_id.
Hash entries deleted	Fixup delete hash entry for the app_id.
Hash entries updated	Fixup update hash entry for the app_id.
Hash miss errors	Fixup hash entry delet error for the app_id.
DNS offset error	The initial pointer for the domain name label length is out of range
DNS offset error2	The pointer for the domain name label length is out of range
ICMP packet received	Fixup received ICMP packet.
ICMP error packet received	Fixup received ICMP error packet.
Fixup IPCP msgs sent	Fixup buddy connection manager sent IPCP to all other NP for pinehole creation.
app_id allocation error	Attempt to allocate app ID returned NULL.
app_id empty error	Errors while attempting to free app ID.
No packet in chain	Encountered a null pointer when going through packet chain.
NAT app fixup response error	Any error during NAT fixup.
DNS loop error	Number of bytes read greater than packet length.
DNS response count error	Not currently used.
DNS response wrong dnsdp	DNS data pointer beyond data range.
DNS response no buffer	After getting a DNS_TYPE_A response, buffer allocation for NAT failed.
DNS connection mismatch	Not currently used.
DNS query parse error	The check of the domain name returned a null pointer.
DNS response parse error	Response domain name label length error.
DNS response rr parse error	Advertised RRs check error.
DNS response packet length error	The response exceeds the DNS proxy maximum size.
DNS query packet length error	The request exceed the DNS proxy maximum size.
DNS query l5 offset error	Not currently used.
DNS query packet type error	Not currently used.
DNS close error	Fixup transmission queue received a message not for close (it should only get close messages).
DNS NAT connection mismatch error	Fixup transmission received naf response but the proxy entry is no longer valid.
DNS cptr range error	When processing NAF, the response size was over the maximum size limit.
DNS proxy conn is in use	DNS proxy entry been used when new data come in on this proxy
DNS protocol ID error	Fixup policy protocol ID did not match the fixup DNS ID.
Other error	Unknown packet type.
Pinhole creates	Pinhole create error for one of a variety of reasons.

Tracker Links	Link data connection with control connection.
Fixup IPCP msgs recd	Received IPCP message at rx ring (now only used for buddy connection create requests).
Drop: Fastpath Queue Full (DNS)	Fixup attempt to send packet to the FASTTX failed.

show np 1 me-stats -shttp

Displays statistics related to HTTP message processing. The output includes statistics per blade and, if specified, per context. If no context is specified in the command invocation, the context-related stats in the output reflect activity for all contexts.

The context specific statistics are all those listed in the output below that fall under the "Context 2 Statistics" line, in the sample output.

Note that the counters in the "show stats http" command are the same as those in the per-context "show np 1 me-stats -shttp" output.

Sample Output

```
ACE30001/RLB_test# show np 1 me-stats "-shttp -v"
HTTP Statistics (Current)
--------------
Unknown msgs received:                          0            0
Data rx msgs received:                   76796579         1027
TCP proxy rx msgs received:                     0            0
Ack trigger rx msgs received:                   0            0
TCP event rx msgs received:                     0            0
Dest decision tx msgs received:          76329650         1027
LB dest decision tx msgs received:              0            0
Close tx msgs received:                    163041            0
Inspect allow tx msgs received:                 0            0
Inspect drop tx msgs received:                  0            0
DRAM blocks read:                       142076143         1955
Buffers dropped:                         76794777         1027
Regex states read:                     1551641642        21406
Unproxy cancellations:                          0            0
Redundant closes:                               0            0
Internal errors:                                0            0
Conn mismatch errors:                          12            0
Exception with close:                           0            0
Dest errors:                                    0            0
Total Packet count (Tx  & Rx):           153289270         2054
Stop regex:                                     0            0
(Context 2 Statistics)
Parse result LB msgs sent:               163351098         2153
Drop: LB queue full:                            0            0
Parse result Inspect msgs sent:                 0            0
Drop: Inspect queue full:                       0            0
TCP data msgs sent:                             0            0
TCP queue full:                                 0            0
SSL data msgs sent:                             0            0
SSL queue full:                                 0            0
TCP fin msgs sent:                              0            0
TCP rst msgs sent:                              0            0
SSL fin msgs sent:                              0            0
SSL rst msgs sent:                              0            0
Bounced fin msgs sent:                          0            0
Bounced rst msgs sent:                          0            0
Unproxy msgs sent:                              0            0
Drain msgs sent:                          2687954            3
```

```
Reuse msgs sent:                              0            0
Particles read:                        76338963         1027
HTTP requests:                         65708556          928
Reproxied requests:                           0            0
Headers inserted:                             0            0
Headers removed:                              0            0
Headers rewritten:                            0            0
HTTP redirects:                               0            0
HTTP chunks:                                  0            0
Unproxy conns:                                0            0
Pipelined requests:                           0            0
Pipeline flushes:                             0            0
Whitespace appends:                           0            0
Response entries recycled:                    0            0
Second pass parsing:                          0            0
Vserver mismatch errors:                      0            0
Analysis errors:                              0            0
Static parse errors:                          0            0
Max parselen errors:                          0            0
Resource errors:                              0            0
Invalid path errors:                          0            0
Bad HTTP version errors:                      0            0
Header insert errors:                         0            0
Header rewrite errors:                        0            0
Invalid policy errors:                        0            0
Invalid rserver errors:                       0            0
Recycled requests:                        63582            0
```

Notes

This table provides details on some of the statistics shown in the output.

Field	Description
Buffers dropped	This is incremented by http for a number of cases including normal case. For example when the buffer chain is dropped normally, this counter is incremented. There are additional error counters incremented if the drop occurs because of error.
Conn mismatch errors	When this increments without incrementing "Exception with close", it indicates a non-fatal error of receiving a msg for a proxy ID which had either been closed or reused with a new sequence number. It can occur for a number of reasons, the most common being that the connection was closed while data was being received (not unusual since HTTP gives priority to close msgs). When this occurs, the message is simply dropped. It is fairly common to see this stat increment during any kind of stress, and does not by itself indicate a bug.
Exception with close	This is the fatal version of conn mismatch error, which results in HTTP sending a RST to TCP on both sides. In this particular instance, the number exactly matches the number of vserver mismatch errors, so it's normal during a config update.
Stop regex	Could not find the regular expression. For example, if SSL session-ID stickiness is configured, and the client or server hello packet does not contain a session-id.

Parse result LB msgs sent	The HTTP Microengine has generated a parse result for a loadBalance decision (header, cookie, or URL string), and sent the results to the LoadBalance application on the Xscale to use in the (L7) load balancing decision.
	
	These L7 loadbalance match classes are configured as:
	
	`class-map type http loadbalance match-any/match-all CLASS-NAME`
	`match http ...`

	and are used in defining an L7 loadbalance policy.
Drop: LB queue full	The Parse result message above could not be delivered to loadBalance because there was no room on LB's receive queue. LB is not draining its queue fast enough. This indicates a problem and needs to be investigated with Cisco TAC.

Parse result Inspect msgs sent	The HTTP Microengine has completed parsing for an HTTP Inspect rule, and sent the results to HTTP Inspect application on the Xscale to be logged/permitted/reset.
	
	These L7 inspect match classes are configured as:
	
	`class-map type http inspect match-all/match-any CLASS2-NAME`
	`match header/url/content...`
	
	and are used in defining an L7 HTTP inspect policy.

Drop: Inspect queue full	The Parse result message above could not be delivered to HTTP inspect because there was no room on inspect's receive queue. Inspect is not draining its queue fast enough. This indicates a problem and needs to be investigated with Cisco TAC.
TCP data msgs sent	While parsing data, the HTTP microengine needed to send either parsed data for forwarding or TCP flags to the TCP microengine to keep the TCP connection going. This is normal.
TCP queue full	Messages could not be delivered to TCP because its receive queue was full. This indicates a problem and needs to be investigated with Cisco TAC.
SSL data msgs sent	While parsing data, the HTTP microengine needed to send either parsed data for forwarding to the SSL microengine to keep the connection going. This is normal.
SSL queue full	Messages could not be delivered to SSL because its receive queue was full. This indicates a problem and needs to be investigated with Cisco TAC.
SSL fin/rst msgs sent	If the HTTP module on the ACE detects a condition under which the TCP connection must be closed or reset, those messages are sent and logged here. These can be normal.
Bounced fin/rst msgs sent	If HTTP gets a FIN/RST indication in the TCP connection prior to beginning parsing of an HTTP request, and no "other side" connection has been opened to the ultimate receiver of the request, then the FIN/RST is "bounced back" (properly returned) to the sender, and the existing connection is closed. An example of this (although not the only possible one) is a client who sends SYN, SYN/ACK, RST to the ACE.
Unproxy msgs sent	Requests by the HTTP module that a connection be unproxied. An unproxy request will be sent when the HTTP response header is complete (ending the completed request/response transaction) or when the response data in its entirety ends, for chunked encoding. If there is a pipelined request when the response is completed, the connection cannot unproxy. In this (the pipelined) situation, this stat will still go up, but the unproxy will later be cancelled. Note that unproxy is never attempted for SSL traffic (nor for server-conn reuse traffic in A1.x).
Drain msgs sent	A response has been received and completed for an HTTP request. Anytime persistence-rebalance or server-conn reuse is enabled, a drain message is sent at the end of the response. This is because there's no way to tell without checking whether there is a pipelined request waiting to be parsed.
Reuse msgs sent	Number of times the HTTP process requests that a connection be placed in the reuse pool. This increments every time a server-connection reuse connection is freed and HTTP requests

	that it be returned to the pool. (That is, there need not be this many simultaneous connections.)
Particles read	Particles are ?convenient pieces? of packets for processing. This is an internal counter unlikely to be useful outside of DE.
HTTP requests	Total HTTP requests received by HTTP for parsing, pipelined or not. This counts only those HTTP requests actually received by the HTTP module. If a connection is configured using "persistence-rebalance", this count will include all HTTP requests received on that connection since each one must be parsed. Connections configured for "server-conn reuse" also require parsing of every HTTP request because a server-side connection can only be reused after it has transmitted the last byte of the response. Parsing of the response, and hence also of the request, must be done to make this determination. However, if neither "persistence-rebalance" nor "server-conn reuse" is configured only the first HTTP request on a connection will require parsing. As of ACE 2.0, this count includes non-HTTP requests (e.g., SIP, skinny, Radius, generic TCP/UDP) that flow through the HTTP module.
Reproxied requests	HTTP requests which must be parsed that are received on a connection which has previously been unproxied require that the connection be reproxied. As an example, connections with "persistence-rebalance" configured can reproxy, since a second request on an already unproxied connection requires parsing by HTTP to see if the connection should be rebalanced. This also occurs for server-conn reuse?each request is parsed, and the response must be parsed also, to know when the server-side connection can be returned to the reuse pool.
Headers inserted	HTTP headers inserted into the HTTP request or response by the HTTP module. This includes both the "Connection: Keep-Alive" header for a request being sent to the server over connections configured with "server-conn reuse", and a header inserted using the header insert feature.
Headers removed	HTTP headers removed by the HTTP module from requests/responses. In ACE 1.x this is limited to removing "Connection: Close" headers on HTTP requests coming from the client which will be forwarded to the server using ?server-conn reuse?. (Telling the server to close these connections would defeat the purpose.) In ACE 2.0, header removal of user-specified header names is also supported.
HTTP chunks	"Chunks" of HTTP data received by the HTTP module. A discussion of chunked encoding may show up here someday.
Unproxy conns	A request by HTTP to unproxy a connection was successfully completed.
Pipelined requests	A valid subsequent HTTP request arrives prior to the arrival of the response to the previous HTTP request. For example, a second GET may arrive from the client before the 200 OK response to the first one is received. Parsing of the second request in the pipeline is deferred until the response to the first request has been received and processed?that is, this request is pipelined.
Pipeline flushes	Pipelined data (as opposed to a single HTTP request) is sent (flushed) to TCP or SSL if the server closes the connection prematurely using FIN/RST. In this case, all the pipelined data is sent, rather than waiting for each HTTP response in an orderly fashion and parsing the pipelined requests.
Whitespace appends	The pipelined "request" is not an actual HTTP request but is (legal, no-op) whitespace.
Response entries recycled	These are HTTP responses received, that are reused to send HTTP requests. This seems like a reasonable approximation of HTTP requests that are sent over existing connections empirically in my tests of "server-conn reuse". Note however that this stat isn't specific to the

	server-connection reuse feature, since persistent connections may also send multiple requests over the same backend connection even if reuse is disabled, if the same real server is chosen or these subsequent requests. If an unproxy had occurred between requests, this stat would not go up.
Second pass parsing	HTTP may need to parse twice, once for loadbalancing and again for HTTP inspection. This indicates a second pass of parsing on the same HTTP request or response.
Vserver mismatch errors	This and the following statistics indicate errors. In general, the connection will be reset by errors indicated here.
Analysis errors	This counter catches errors in the HTTP parser itself. This indicates a problem and should be brought to the attention of Cisco TAC.
Static parse errors	Either the client or the server data parsed by the HTTP module did not conform to correct HTTP format, and the parse was aborted. This could indicate a problem and should be investigated with Cisco TAC.
Max parselen errors	The HTTP module reached the end of the configured maximum parselength without finding a match for the desired regular expression. The connection is not necessarily reset as a result of this.
Resource errors	A buffer or other internal resource required by the HTTP module was not available. This is never expected and needs to be investigated with Cisco TAC.
Invalid path errors	This counter may increment as a result of a race condition during processing. The "path" referenced is the path the buffer will take within the ACE. This is an unexpected event that is unlikely to occur.
Bad HTTP version errors	Only HTTP version 1.x is expected?other versions cause this counter to increment.
Header insert errors	This counter indicates that an HTTP header could not be inserted. If this counter increases in lockstep with "Resource errors" above, the failure is due to a problem getting resources?but this is not always the case. This indicates a problem and needs to be investigated with Cisco TAC.

show np 1 me-stats -sicm

Displays statistics regarding activity of the ICM (Input Connection Manager) micro engine. This information can be useful for troubleshooting; in particular, if the first three counters in the output are accruing errors or if messages are not shown to be received in the output, you should pursue the matter with Cisco TAC.

This command also displays output for the CM Close process, which is separate from ICM. It is responsible for closing connections. The stats for CM Close are prefixed in the output with the "Close" qualifier.

Any stats prefixed with ?Reuse? are concerned with ?server-conn reuse? connections. Also, please note that a server-side connection is not in the reuse pool if it is currently associated with a client-side connection. Only server-side connections which are immediately available for use are in the reuse pool. A server-side connection remains in the pool until the server closes it, or the ACE times out for inactivity using the standard timer.

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-sicm -v"
```

```
ICM Statistics (Current)
--------------
Errors:                                    23            0
Frames Received:                         6064076        10
Drop [unknown msg]:                           0            0
IPCP Received:                                9            0
Embryonic Hit Received:                       0            0
Close Receive:                            89997            0
Close Drop unknown msg:                       0            0
Close Errors:                                 0            0
Close Connection timeout:                  7060            0
Close IPCP send stat:                         0            0
Close IPCP recv stat:                         0            0
Encaps Miss Success stat:                     0            0
Encaps Miss Error stat:                       0            0
Close No interface on connection:             0            0
Close connection [Interface down]:            0            0
Close Zero CID in message:                    0            0
reap messages received:                       6            0
reap messages processed:                      6            0
reap IPCP Reply:                              6            0
Reuse link update conn invalid error:         0            0
Reuse link update conn not on reuse erro      0            0
Reuse conn remove not on head error:          0            0
Drop [Next-Hop queue full]:                   0            0
Close Error not in hash:                      0            0
Invalid reap messages:                        0            0
If lookup error:                              4            0
encap lookup error:                           0            0
Replicate bulk sync done sent to HA:          4            0
Replicate connection if xlate error:          0            0
Replicate connection update existing:     16538            0
Drop [buffer threshold limit]:                0            0
Drop [Buddy connection alloc error]:          0            0
Close Connection validation error:         1980            0
UDP Chaser sent, conn miss:                4102            3
UDP Chaser sent, partial conn:            46513           29
Reuse shutdown free connection error:         0            0
Reuse connection already freed error:         0            0
(Context ALL Statistics)
Transmit -> fastpath:                     17991            0
Transmit -> TCP:                           9705            0
Transmit -> OCM:                          24937            0
Send   -> LB_L4:                          24910            0
Send -> Other IXP:                         1694            0
Drop [redundant]:                             0            0
Drop [ACL deny]:                              0            0
Drop [Connection RL]:                         0            0
Drop [CP Connection RL]:                      0            0
Drop [Proxy RL]:                              0            0
Drop [SSL RL]:                                0            0
Drop [Connection Rate RL]:                    0            0
Drop [Inspect Rate RL]:                       0            0
Drop [IF FT Standby]:                   5944642           10
Drop [ICMP Hard Error]:                       0            0
Drop [ICMP Redirect]:                         0            0
Drop [ICMP Error IP Mismatch]:                0            0
Connection [Inserts]:                     92151            0
Connection [Deletes]:                    151670            0
Connection [Modifies]:                        0            0
Proxy [Inserts]:                              0            0
Proxy [Deletes]:                          69284            0
IPCP Sent:                                    9            0
CP Init Received:                          5945            0
```

```
Invalid conn miss TCP flags:                    37              0
RPF check Error:                                 0              0
Route lookup Error:                            387              0
MAC Lookup Error:                                0              0
To CP - My mac check Error:                      0              0
Bridged - My mac Error                           0              0
BVI invalid/down Error                           0              0
Classify Error:                                 37              0
Transmit Encap Miss Msg stat:                    0              0
Drop [Encap Miss Msg stat]:                      0              0
Close Connection with invalid proxy:             0              0
Pinhole deletes:                                 0              0
Tracker Unlinks :                                0              0
Connection Reuse Add Errors:                     0              0
Connections Removed From Reuse Pools:            0              0
Connections Added To Reuse Pools:                0              0
Replicate Connection encap lookup error:         0              0
Replicate Connection MAC lookup error:           0              0
Replicate connection sent:                       1              0
Replicate connection msg to other ixp:           0              0
Replicate connection recv L4:                    0              0
Replicate connection recv LB:                24910              0
Replicate connection recv buddy:                 0              0
Drop [Replicate conn buddy - no control          0              0
Close IPCP errors:                               0              0
Close connection tracker not found error         0              0
```

Notes

Field	Description
Errors	ICM tried to pull a message off its input queue, and the message was of ?null? type, which should not occur. ICM counts the attempt, skips this ?message?, and waits for more input.
Frames Received	Pulled a message off the queue which is either of valid type (packet or IPCP msg) or not. If this stat is incrementing, ICM is pulling/ processing messages from its input queue (which is good).
Drop [unknown msg]	Not a valid message (packet or IPCP msg) on the main queue [OR] this seemed to be a valid packet/ IPCP msg, but was not of a valid type when ICM tried to parse it. Basically, ICM pulled something totally uninterpretable off its input queue. This should never increment.
IPCP Received	IPCP is the messaging protocol used between components on the ACE blade. For example, a message from the CP (control processor) to one of the network processors (IXP0) can be sent via IPCP, or from one of the network processors to the other (IXP0 to IXP1) using IPCP.
Embryonic Hit Received	An embryonic connection is one which has not finished the handshake?that is, one that is not ready to send data. Typically, you worry about SYN attacks?SYNs to which the SYN/ACK response receives no reply, sucking up resources until the timeout occurs. In practice, there is apparently no place in the ACE 1.x codebase where this stat is incremented, so I'm not sure what it measures. Perhaps in ACE 2.0? If you see it increment, I'd be interested in the code version you're running.
Close Receive	The CM Close (Connection Manager Close) process has received a message to process,

	usually requesting that a connection be closed (which is a good thing.) This can be from TCP, from NAT ? This is exactly analogous to ?Frames Received? for ICM (above).
Close Drop unknown msg	Couldn't parse this message. Again, this is the CM Close analogy for the ?Drop Unknown? stat for ICM above.
Close Errors	CM Close went to close some sort of connection (proxied (see the ?shttp stats for a discussion of this), and couldn't find the connection to close. This might occur occasionally because of timing windows, but a steady drain is not a good thing?particularly if we are apparently leaking either connections or proxies.
Close Connection timeout	CM Close is closing a connection because it has timed out. This can be normal?or indicate a network issue?or indicate a bug.
Close IPCP send stat	CM Close is closing a secondary connection (data channel associated with a control channel (e.g., FTP data channel). Since the hash determining which IXP receives the traffic for a connection is based on (source port ^ dest port), there is no guarantee that the data and control connections will be kept on the same IXP. If the two connections are on different IXPs, the other IXP is notified via an IPCP message. Sending that message is counted here.
Close IPCP recv stat	Ideally, this would be the receive side of the above stat. In practice, the ACE 1.x image seems never to increment this stat.
Encaps Miss Success stat	ICM did not find an encaps entry (the layer 2 information necessary to forward a packet). This does not cause ICM to drop the packet.
Encaps Miss Error stat	ICM could not forward the packet because of its failure to find a suitable encaps entry - this may mean it does not know the src mac or the dest mac. The packet is dropped.
Close No interface on connection	Connections are closed by CM Close where the interface information in the connection corresponds to no known interface.
Close connection [Interface down]	Ideally, CM Close would shut down connections on an interface if the interface went down. In practice, this stat is not used in ACE 1.x code.
Close Zero CID in message	CM Close received a close request for a connection which it could not find in the global connection database. This should not occur. Connections are normally closed because of a connection-related event?FIN, RST or timeout. Connections can be released/removed because of an external event, for example an rserver becomes unavailable because a probe fails, an interface goes down, a serverfarm is deleted,?or whatever. These connections should be removed to free the associated resources. Connections removed because of an external (non-connection) event are ?reaped? in the dataplane in response to notification of the event observed in the control plane.
reap messages received	CM Close receives messages from the CP (Control procdessor) to close or ?reap? the connections associated with some entity.
reap messages processed	CM Close places the received reap messages on its own internal queue to unburden IPCP while the messages are being processed. If reap messages are being received but not processed, this indicates a problem with CM Close handling of these messages.
reap IPCP Reply	When CM Close receives reap messages, it may be required to acknowledge them to the sending process. If so, it will send a reap IPCP reply.
Reuse link update conn invalid error	ICM went to add or remove a connection from the reuse pool (those connections kept open with the server) and was unable to find a valid connection associated with this request. Not good.

Reuse link update conn not on reuse erro	ICM went to add or remove a connection from the reuse pool, but this connection is not marked as suitable for reuse. This also is not good.
Reuse conn remove not on head error	ICM went looking for a connection to remove from the reuse pool, and didn't find it on the expected list. This is fairly obscure, and not good.
Drop [Next-Hop queue full]	ICM wanted to send a packet to another component (fastpath, LoadBalance,?) for processing, but couldn't get it on that component's queue. This is not good, indicating a problem in the destination component. Check the "Transmit" and "Send" stats below to see who ICM could be trying to talk to.
Close Error not in hash	CM Close should always be able to find the connection to close using the hash. Not good.
Invalid reap messages	Similar to the reap messages above; this indicates a message that was received but could not be processed because the parse didn't make sense. CM Close doesn't even bother trying to process these. This stat should not increment.
Replicate bulk sync done sent to HA	Bulk sync happens as soon as a redundant card comes up. All the connection in the now active box are replicated to the new redundant pair and on completion bulk sync flagis sent, which the redundant pair uses to notify the HA modify. This is a notification to say that both the boxes are in sync state and so now its good to switchover if required.
Replicate connection if xlate error	This stat is the place to start if you have a problem replicating connections. Idmap for interface id returned error in the standby. This error is incremented on the standby when the IFID of the interface on the active blade does not have a corresponding mapping on the standby blade. The packet replicated does not have an interface associated with it. This mapping of Object Id on the Active to Object Id on the Backup is known as the "idmap". The Object can be an interface, an rserver, anything configured on the ACE blade which is used in replicating the connection.
Replicate connection update existing	Connection has been already created in the standby and the periodic replication packets are used to just update the connection, say to update reproxy fields.
Drop [buffer threshold limit]	ICM keeps a certain number of buffers free to accommodate existing connections, and so is dropping a new connection. This may indicate we're over-subscribing the blade, or it may be a buffer leak.
Close Connection validation error	This gets incremented when cm_close is trying to close a connection which has been freed already.
UDP Chaser sent, conn miss	If a UDP conn_id is in the process of being created when a second.
UDP Chaser sent, partial conn	Packet is received on the connection, then the packet is forwarded to ocm using the partially established connection, and one of these two counters is incremented (depending where in conn setup the conn miss was detected). This is normal operation and is part of the fix for CSCsu42225.
Reuse shutdown free connection error	ICM is freeing a connection, and is considering adding this to the reuse pool. However, this doesn't appear to be a valid connection. Not good.
Reuse connection already freed error	ICM is trying to get a connection to remove, and it Turns Out not to be on the list. This is an internal housekeeping error, and is Not Good.

Note: The stats below this are kept per-context	
Transmit -> fastpath	Not L4 or L7, but probably bridged
Transmit -> TCP	ACE is terminating this connection
Transmit -> OCM	Routed packet
Send -> LB_L4	Load balanced
Send -> Other IXP	Hash mismatch (ICMP error pkts) ICM can receive ICMP error packets from stations on the network returning errors for some packet they received. The internal information in these records need to be translated into what the client expects (If the client sent traffic to the VIP, it will not understand either and IP header or an ICMP error payload from the server address.) The connection records for these packets may be on the other IXP so, to NAT the contents properly, the packets may be sent on to the other IXP for forwarding.
Drop [redundant]	This is incremented when ICM went to set up a connection structure, and was unable to do so. (obviously) In general it is due to second packet or following try to create a redundant connection and find there are already connections with same hash in table.
Drop [ACL deny]	
Note: In the following counters, RL stands for "Rate Limit". Drops of connections in excess of the listed Rate Limit are shown in this counter and the other RL counters in the output.	
Drop [Connection RL]	These are drops due to exceeding the connection rate setting in ?show resource usage?
Drop [CP Connection RL]	These are drops due to exceeding the mgmt-traffic rate in ?show resource usage?
Drop [Connection Rate RL]	These are drops due to exceeding the syslog rate (including both the DP and the CP rates) seen in ?show resource usage.?
Connection [Inserts]	Number of connections that were created
Connection [Deletes]	Number of connections that were deleted.
Invalid conn miss TCP flags	Indicates ICM TCP normalization failures
RPF check Error	This packet failed ICM's Reverse Path Forwarding (RPF) check. Count it and drop the packet.
Route lookup Error	ICM couldn't find a route. Drop the packet. This is apparently only incremented on traffic that is being sent from the ACE itself, like probes, pings, and so on.
MAC Lookup Error	Fail on a lookup of a local reverse-encaps. Drop it.
To CP - My mac check Error	Drop any unicast packets sent to the CP (probe responses, telnet?) if the destination mac is both unicast and NOT the ACE mac.
Bridged - My mac Error	Drop any DHCP packets hitting an LB VIP in bridged mode.
BVI invalid/down Error	Drop it.
Classify Error	Indicates ICM TCP normalization errors

Pinhole deletes	CM Close removes the ?pinhole? (secondary data channel) along with the control connection when closing connections which use multiple channels (e.g., FTP). This is appropriate, expected behavior.
Tracker Unlinks	Linked connections (e.g., the FTP data and control connections) are tracked in the tracker database. When CM Close closes the connection, the database entries are unlinked and freed?which is counted here. This is appropriate, expected behavior.
Connection Reuse Add Errors	This counter and the other connection counters below it track server-conn reuse; see also **show np 1 me-stats ?shttp**
Note: The following counters track connection replication in FT (Fault Tolerant) configurations.	
Replicate Connection encap lookup error	The connection is dropped on the STANDBY because the encap id for the incoming interface is missing.
Replicate Connection MAC lookup error	ICM on the STANDBY could not find an encap for the source mac in the replicated connection. Drop.
Replicate connection sent	The CM_Close process on the ICM ME on the ACTIVE sent a UDP connection replication packet to the STANDBY. Each packet contains one connection, which is two connection records. A specific connection is sent every three minutes while the connection is active and being replicated.
Replicate connection msg to other ixp	The ICM ME sent a message to the other IXP on the ACTIVE ACE. Basically, when buddy connections are replicated, buddy information describing the data connection will be sent to the IXP that has control connection. The control connection information is gathered and the replication message is sent to the IXP on the STANDBY that has the control connection. Data connection setup will take place exactly in the same way an active would have setup the buddy connection. This is the case where control connection is in the local IXP, whereas data channel connection is in the other IXP. So control tuple is filled in the packet and sent to other IXP, which would then replicate it to standby.
Replicate connection recv L4	At some point this ACE was in Standby. These are connections which were made with only ICM input (ACL lookup, routing, bridging). It is incremented for non-slb connections replicated from the active to standby.
Replicate connection recv LB	SLB connections replicated from active to standby.
Replicate connection recv LB	A connection?either L4 or L7?made by Load Balance (LB) is replicated. ICM on the STANDBY receives the connection info, which includes the serverfarm ID and the rserver ID, and forwards it to the LB module where the idmap (see above) is used to select the same serverfarm/rserver pair on the STANDBY. Note that these connections are unproxied before they are replicated.
Replicate connection recv buddy	The STANDBY ACE has received a control connection and its corresponding buddy info. Any data channel connection replicated from active to standby.

The central operating principle is that in Connection Replication, connection information is replicated from CM_Close on the ACTIVE to the ICM in the standby. CM_Close replicates the connections either because ICM has requested this for a new connection or because CM_Close is walking the connection database

updating existing replicated connections on a periodic basis. The information is used in the standby to set up (or update) these connections following the same logic that was used to create the original connection on the ACTIVE.

show np 1 me-stats -sidle

The idle stats for each DP module are incremented every time that the module polls its queue for messages and has no message waiting. If there is no message, it increments this counter, then goes to sleep for a period of time before polling the message queue again.

For ACE 2.x, the output lists two numbers. The first number is a historical count from system boot (and wraps for most counters twice a day). The second is a count of idle returns in the past second. For most modules, this is around 100k per second.

Sample Output

```
ACE30002/Admin# show np 1 me-stats  -sidle
Queue Idle Statistics (Current)
--------------
RECEIVE:                                0x19db08cf
FASTPATH:                               0x34e33345
SLOWTX:                                 0x17fb5e47
REASSEMBLY:                             0xf893cc5c
TCP_RX:                                 0x18851467
HTTP:                                   0x3da523c9
IH_RX                                   0x11e80eab
SSL_ME:                                 0x0a2e9b57
CM_CLOSE:                               0x13b83417
X_TO_ME:                                0x7dbadc02
FIXUP:                                  0xf6f38374
OCM:                                    0x0a3df3d5
TCP_TX:                                 0xd34a2618
ICM:                                    0x18a02563
```

show np 1 me-stats -slb

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-slb -v"
LB Perf stats at address 0x82e05000
LB Perf stats at address 0x82e05000
LB Statistics
--------------
(Context ALL Statistics)
Config Version Mismatch:                   0            0
No Policy:                                 0            0
No Policy Match:                           0            0
No Real Server:                            0            0
ACL denied:                                0            0
L4 LB Decisions:                           0            0
L4 Rejected Conns:                         0            0
L7 LB Decisions:                        9708            0
L7 Rejected Conns:                         0            0
FT Idmap Lookup Failures:                  0            0
```

```
Proxy Close Drops:                          191             0
Misc Drops:                                   0             0
L4 Close Before Process:                      0             0
L7 Close Before Parse:                        0             0
Close For Valid Real:                         0            60
Close For Invalid Real:                       0             0
Max Parse Len Rejects:                        0             0
L7 Parser Error Rejects:                      0             0
Out of Memory Rejects:                        0             0
Config Mismatch Rejects:                      0             0
HA Send Failure:                              0             0
HA Packets Sent:                              1             0
HA Entries Shared:                            0             0
HA Received:                                  0             0
HA Packets Received:                          4             0
HA Entries Dropped:                           0             0

Num Stolen For Reuse:                         0             0
Num Active Sticky Entry:                      0             0
Num Active Reverse Sticky Entry:              0             0
Active Conn Count:                            0             0
Free Sticky Entry Count:                 546097             0
Num Grp or Timeout Nodes:                     2             0
Static Entry List Count:                      0             0
Num Entry Configured:                    546097             0
Prev Resources Req:                      546097             0
Drop Max Remote Stky:                         0             0

RTSP sessions allocated:                      0             0
RTSP sessions failed:                         0             0
RTSP sticky entries added:                    0             0
 SIP sessions allocated:                      0             0
 SIP sessions failed:                         0             0
 SIP sticky entries added:                    0             0

Free Proxy Mapping:                       32768             0
Alloc Proxy Mapping:                          0             0
Alloc Proxy Mapping Failed:                   0             0
Release Proxy Mapping:                        0             0
```

Notes

Field	Description
The following statistics are for all contexts.	
Config Version Mismatch	Vserver version on the proxy info or in the vserver state does not match that expected for the L7 connection. This occurs when new (reconfigured) vserver info is passed to the dataplane. LB is unable to return a decision, and the connection is close or not formed.
No Policy	LB was unable to find a policy associated with this L4 connection. (no vserver, or no default policy). The clientmap trees are not part of lb bank, this is a binary tree consisting of match source addresses generated by lb fabric using the L7 match source address configurations. This gets downloaded into common mempool address space in dram and LB uses this to perform policy selections. So the "no policy" can happen as a result of LB not finding the right lb policy and in turn result to default if one configured or else will drop the connection.
No Policy Match	LB was unable to find a policy which satisfies all the conditions of the L7 connection.
No Real Server	Policy and vserver were OK, but there was no acceptable rserver available for LB to send the connection to.

ACL denied	The configured policy associated with this connection indicates it should be dropped. This can occur for either an L4 or an L7 policy.
L4 LB Decisions	LB loadBalanced, stuck, or forwarded an L4 connection.
L4 Rejected Conns	LB rejected an L4 connection. When this is incremented, sometimes another counter is incremented (in -slb stats). In all cases a drop message is sent to the xtomelo queue with the drop reason. To see the counters, so 'show np 1/2 me-stats -slb'. To see the queue reasons, do 'show np 1/2 me-stats -qxtomelo'. Here is a list of these counters and drop reasons: The **show np 1/2 me-stats -slb** counters: • No Policy • ACL Denied • Drop Max Remote Stky • No Real Server show np 1/2 me-stats -qxtomelo reasons follow. These reasons appear in the output as follows: `[LB Dest Decision / Seq Mismatch]` Since there are not necessarily counters for these, the queue has to be inspected while the particle is on the queue. Therefore you may have to run the command many times to catch a low-probability event. Here are the associated drop reasons: • Max Capacity • VIP Not In service • Seq Mismatch • Error • Policy Drop • No Policy
L7 LB Decisions	10 LB loadBalanced, stuck, or forwarded an L7 connection.
L7 Rejected Conns	LB rejected an L7 decision for any of a multitude of reasons, including: • no available real server • acl deny • VIP not inservice • VIP config version mismatch (similar to HTTP below, but for LB) • any of the http errors below ♦ connection invalid ♦ no valid policy ♦ error receiving sticky info from other IXP ♦ exceed max capacity on rserver
Proxy Close Drops	Due to a late arriving response that has come after hearing a msg from connection mgr to close the connection.
The following stats are HTTP stats. Incrementing ANY of the HTTP stats will be accompanied by an increment in the "L7 Rejected Conns:" stat above	
Max Parse Len Rejects	HTTP has parsed the maximum number of characters permitted without finding the specified expression. Packet rejected.

L7 Parser Error Rejects	HTTP has attempted to parse the packet, and has found a non-acceptable character. This occurs when parsing an HTTP method, where only graphical characters or whitespace is allowed. Packet is rejected.
Out of Memory Rejects	HTTP has attempted to get memory for packet manipulation, and has failed. This error is sometimes, but not always, associated with the inability to insert data (e.g. Cookie or Header) into a packet. Packet is rejected.
Config Mismatch Rejects	HTTP has detected a change in the vserver information it is currently using (stored in its local memory) and the vserver information it reads from global (DRAM) memory. This usually implies a configuration change between now and when HTTP began to parse the connection. This is too inconsistent to continue. Packet is rejected.
The following stats are HA stats for HA data sent directly from the loadBalance process on the NP. This is "replicate sticky" related data, not heartbeats or other HA configuration data.	
HA Send Failure	This is an internal failure trying to send a message to an HA peer. These messages are required for sticky data replication/purge to the HA peer
HA Packets Sent	This is a message successfully sent to the HA peer for sticky data replication/purge. Note: Every attempt to send a message will result in either "Packets Sent" or "Send Failure" incrementing.
HA Entries Shared	The number of sticky entries sent (successfully, as above) to the HA peer. (There can be multiple sticky entries per HA packet).
HA Received	HA packets dropped upon receive because they are not HA packets or they are an unrecognized version of these packets. There is no way to determine what type of message this should be. This should never increment.
HA Packets Received	Packets successfully received by HA, with a valid message type. We attempt to process these messages.
HA Entries Dropped	Multiple sticky entries can be received in each HA packet. If the entry cannot be added to or removed from the sticky database on the receiver, this counter is incremented. Reasons for this include: 1) can't resolve the mapping on the receiving ACE to the sticky group or real server that is refered to in the sticky entry 2) the sticky group this entry is part of is not active on this ACE 3) the attempt to insert the sticky entry into the database failed This counter could increment, but it would not be good.
The following stats are HA stats for HA data sent directly from loadBalance on the NP. This is "replicate sticky" related data, not heartbeats or other HA configuration data.	
Num Stolen For Reuse	Same as "Total sticky entries reused prior to expiry" from "show stats sticky" (summed over np 1 and np 2).
Num Active Sticky Entry	Same as "Total active sticky entries" from "show stats sticky" (summed over np 1 and np 2).
Active Conn Count	Same as "Total active sticky conns" from "show stats sticky" (summed over np 1 and np 2).
Free Sticky Entry Count	This value represents the value of "Num Entry Configured" minus "Num Active Sticky Entry" plus the "Static Entry List Count" value.
Static Entry List Count	Same as "Total static sticky entries" from "show stats sticky" (summed over np 1 and np 2).
Num Entry Configured	The number of sticky entries configured via the resource-class "limit-resource sticky" sub-command. For the module, this is half the value, since there are two NPs.

Drop Max Remote Stky	When the current IXP can't find an entry in the sticky database, it sends a query request to the other IXP. It then waits for the response from the other IXP. The number of pending requests is limited to 8192. Once the threshold is hit, new requests are dropped and this counter is incremented. The number of pending requests is available via LbInspectTool and is called "Pending Remote Sticky Conns".
	
	Here is how to read it:
	
	`# LbInspectTool`
	`[snipped]`
	`Enter:`
	`1dc0 <<<<<<<<<<<<<< Type this (inspect Tables, DRAM, context 0)`
	`[snipped]`
	`Pending Remote Sticky Conns = 0`
	
	The "Pending Remote Sticky Conns" is the counter of interest; drops start when this counter reaches 8192.

Note: All of the above statistics are kept on a per-context basis. Additionally, the HA stats are kept per fault-tolerant group. The sum of all the HA stats per-context is the same as the sum of all the HA stats per-FT group is the same as the total loadBalance/sticky HA stats on the unit.

show np 1 me-stats -snitrox

Displays error statistics related to the activity of the Nitrox cryptographic component.

Sample Output

```
show np 1 me-stats -snitrox
ERR_GP_UNSUPPORTED_CATEGORY                    0
ERR_GP_INVALID_OPERATION                       0
ERR_GP_INVALID_LENGTH                          0
ERR_GP_PARSING_ERROR                           0
ERR_GP_INVALID_MODLENGTH                       0
ERR_GP_INVALID_EXPLENGTH                       0
ERR_GP_INVALID_DATLENGTH                       0
ERR_GP_INVALID_MODULUS                         0
ERR_GP_INVALID_ADDR                            0
ERR_GP_UCODE_AUTH_ERROR                        0
ERR_GP_DIGEST_MISCOMPARE                       0
ERR_GP_CCMP_PKTNUM_MISCOMPARE                  0
ERR_GP_CRC_MISCOMPARE                          0
ERR_GP_MIC_MISCOMPARE                          0
ERR_SSL_UNSUPPORTED_CIPHER                     0
ERR_SSL_UNSUPPORTED_EXPORT                     0
ERR_SSL_UNSUPPORTED_PROTOCOL                   0
ERR_SSL_UNSUPPORTED_AUTH                       0
ERR_SSL_MAC_MISCOMPARE                         0
ERR_SSL_CTX_INVALID                            0
ERR_SSL_VERIFY_DATA_MISCOMPARE                 0
ERR_SSL_INVALID_PADLEN                         0
ERR_SSL_BAD_RECORD                             0
ERR_SSL_SEGMENTATION_ERROR                     0
ERR_SSL_UNKNOWN                                0
```

Notes

The following table provides information on selected counters in the output. Events for the counters not listed here are unexpected and not likely to occur. In most cases, a non-zero value implies fatal error.

Field	Description
ERR_GP_INVALID_LENGTH	Possible causes: • MODEXP OP ? The peer has sent a corrupt certificate. • Other ? This error is unexpected and not likely to occur. Non-zero value implies fatal error.
ERR_GP_PARSING_ERROR	The peer sent a corrupt PKCS envelope.
ERR_GP_INVALID_MODLENGTH	Cert has invalid bitsize on cert key, which is not supported by the nitrox: For example, a 1023-bit key is not supported: RSA Public Key: (1023 bit) Modulus (1023 bit): s/b 1024 Also this from Hai Xiao: For NitroxII modular exponentiation operations: result = (data**exponent) mod modulus the size of modulus in bytes should be 16<= modlength <= 392 The error says modlength is not in range, and indicates some internal error.
ERR_GP_INVALID_EXPLENGTH	The peer has sent a corrupt certificate.
ERR_GP_INVALID_DATLENGTH	The peer sent a corrupt PKCS envelope.
ERR_SSL_MAC_MISCOMPARE	The Message authentication Code value computed was not equal to the supplied value. This error is triggered during SSL record decryption(normal or segmented). This error most likely is the result of the SSL record being corrupted.
ERR_SSL_VERIFY_DATA_MISCOMPARE	The computed Verify Data didn't equal the supplied Verify Data. This error happens when using Client Authentication. The client sends its certificate and verify message(hash of all previous handshake messages). The server generates its own version of the verify message and compares it to what the client sent. If the two don't match, this error is generated. This error can happen if any of the handshake messages get corrupted or the Master Secret is corrupted.
ERR_SSL_INVALID_PADLEN	This check is only done for SSL record decryption, when the Block Cipher is not RC4, i.e. AES, DES, or 3DES. This error is triggered if: SSL_Record_Header.Length - Pad_Length - MAC_Length < 0 Where: SSL_Record_Header.Length is the length field in the SSL Record Header Pad_Length is 8-bit value extracted by decrypting the tail block of the SSL Record. For a normal record this is the last block, and for a segmented record this is the first block(segmented processing pre-pends the last block to the beginning of the record). MAC_Length = 16 for MD5 and 20 for SHA1.

show np 1 me-stats -snormalization

This command and **show np 1 me-stats -snorm** are the same command.

Sample Output

```
ACE30002/Admin#  show np 1 me-stats "-snormalization -v"
Normalization Statistics: (Current)
-----------------------
L3 invalid version:                              0             0
L3 TOS cleared:                                  0             0
[IPv6] invalid extentions:                       0             0
[IPv6] invalid destination options:              0             0
[IPv6] traffic class clears:                     0             0
L3 record route options:                         0             0
L3 timestamp options:                            0             0
L3 SEC options:                                  0             0
L3 L source route options:                       0             0
L3 S source route options:                       0             0
L3 SATNET options:                               0             0
[Drops] L2 invalid DA mac:                       0             0
[Drops] L4 port is zero                          0             0
[Drops] TCP invalid conn miss flags:           613             0
[Drops] TCP invalid flags:                     253             0
TCP cleared urgent pointer:                      0             0
[Drops] TCP urgent pointer denied:               0             0
TCP zeroed reserved field:                       0             0
[Drops] TCP non-zero reserved field:             0             0
[Drops] TCP syn data denied:                     0             0
TCP options are not in LM:                       0             0
TCP no of cleared options:                   29154             0
[Drops] TCP non-syn options on syn:              0             0
[Drops] TCP syn options on non-syn:              0             0
[Drops] TCP no of denied options:                0             0
[Drops] TCP option length wrong:                 0             0
[Drops] fp TCP invalid ack in syn-ack:           0             0
[Drops] fp TCP invalid ack for syn-ack:          0             0
[Drops] fp TCP ack past seq:                     0             0
[Drops] fp TCP window left edge:                 0             0
[Drops] fp TCP window right edge:                0             0
[Drops] fp TCP data past FIN:                    0             0
[Drops] fp TCP FIN has wrong seq:                0             0
[Drops] fp TCP RST has wrong seq:             2448             0
[Drops] fp TCP RST has wrong ack:                0             0
[Drops] fp TCP ack > FIN_ACK exp:                0             0
[Drops] fp TCP exceeded MSS:                     0             0
[Drops] fp IP TTL is zero:                       0             0
Fastpath generated TCP ack:                      1             0
(Context ALL Statistics)
L3 invalid header len:                           0             0
L3 L2/L3 length mismatch:                        0             0
L3 TTL repl:                                     0             0
L3 invalid flags:                                0             0
L3 options removed:                              0             0
L3 invalid options:                              0             0
L3 DF cleared:                                   0             0
L3 invalid address:                            288             0
[Drops] L4 invalid header len:                   0             0
[Drops] icm TCP normalization:                  37             0
[Drops] fp TCP normalization:                 3315             0
```

Notes

The counters from "L3 invalid version" through "L3 SATNET options" are deprecated. The following table contains information on other notable counters.

Field	Description
[Drops] L2 invalid DA mac	Packet dropped with invalid DMAC address (00:00:00:00:00:00).
[Drops] L4 port is zero	TCP/UDP Packet dropped with SRC port number or Dest port number as 0.
[Drops] TCP invalid conn miss flags	PSH flags set. If any other flag is set for a packet that does not have a connection established on ACE, its dropped and this statistic is incremented. Non-syn packets received with no corresponding connection in the fastpath may happen at very high loads. That will cause this counter to increase and may not be an issue.
[Drops] TCP invalid flags	RST, SYN|FIN, and FIN|RST. These would be considered a type of Denial of Service (DOS) attack.
TCP cleared urgent pointer	The TCP Urgent Pointer cleared by default. By default the ACE configuration says to clear this field.
[Drops] TCP urgent pointer denied	The TCP Packet is dropped when urgent pointer is set (if ACE is configured to deny packets with urgent pointer set). Note that the two statistics above are directly related. Either the urgent pointer is cleared and the first statistic is incremented, and the packet is forwarded (by default) OR the user has configured that ACE should drop TCP packets with the urgent pointer set, so the ACE would drop the packet, and the second statistic is incremented. **Note:** There are such pairs of statistics scattered through here where the user can configure the behavior of normalization.
TCP zeroed reserved field	Clear TCP Reserved field (enabled by default).
[Drops] TCP non-zero reserved field	A TCP packet is dropped when the TCP reserved field contains a non-zero value (if configured to deny packet with non-zero reserved field).
[Drops] TCP syn data denied	A TCP SYN Packet is dropped when the SYN packet contains data (if configured to Deny data with syn packet).
TCP options are not in LM	If we cannot fit all the headers (IMPH, L2, L3, and L4 headers) into the Local Memory on the Fast Path (FP) microengine, then we cannot actually check normalization. This would only happen with serious (very large) TCP L3 AND L4 options parameters. If ACE cannot run the normalization sanity checks, the ACE will drop the packet.
TCP no of cleared options	This counter is incremented if a TCP packet is received that has a previously-set TCP option cleared. It is incremented in eight different locations in the ACE source code: • By ICM: ♦ Window Scale Option cleared ♦ SACK Allowed Option cleared ♦ Timestamp Option cleared ♦ By default ACE will clear TCP options • By Fastpath: ♦ Window Scale Option cleared

	♦ SACK Allowed Option cleared ♦ SACK Data Option cleared ♦ Timestamp Option cleared **Note:** By default, the ACE does not TCP options e.g. Timestamp and Window Scaling (WS). Therefore ACE will clear the TCP option. This is Normal operation.
[Drops] TCP non-syn options on syn	Drop the TCP SYN packet with invalid TCP options in SYN packet (eg SACK data).
[Drops] TCP syn options on non-syn	Drop of TCP NON-SYN packets with invalid TCP options. (eg MSS, Window scale, SACK allow).
[Drops] TCP no of denied options	The TCP Packet is dropped when the received TCP options in the packet are in range of configured TCP deny options.
[Drops] TCP option length wrong	The TCP Packet is dropped when the length of received TCP options in packet does not confirm to the standard RFC specified length of respective TCP option field.
[Drops] fp TCP invalid ack in syn-ack	Deprecated.
[Drops] fp TCP invalid ack for syn-ack	A TCP Packet is dropped since the TCP ACK number did not match the TCP sequence number sent in SYN-ACK. A TCP ACK challenge is generated.
[Drops] fp TCP ack past seq	A TCP Packet is dropped since the TCP ACK number in received packet was more than sequence number of packet last sent. A TCP ACK challenge is generated.
[Drops] fp TCP window left edge	A TCP Packet is dropped since the received TCP sequence number was less than last received TCP sequence number. In addition the received TCP sequence number was past one TCP window's worth of data (past 1 window_size behind).
[Drops] fp TCP window right edge	A TCP Packet is dropped since the received TCP sequence number was more than the "window_size" away from last received TCP sequence number.
[Drops] fp TCP data past FIN	A TCP Packet is dropped since a TCP FIN was seen and the ACE does not allow any traffic past the TCP FIN (except for a TCP RST).
[Drops] fp TCP FIN has wrong seq	A TCP FIN packet is dropped because the TCP sequence number was already seen (the TCP FIN retransmit will have the same TCP sequence number and is not dropped).
[Drops] fp TCP RST has wrong seq	A TCP RST packet is dropped if not received with the expected TCP sequence number or already seen TCP sequence number.
[Drops] fp TCP RST has wrong ack	A TCP RST packet is dropped if the TCP ACK number does not match the expected TCP sequence number.
[Drops] fp TCP ack > FIN_ACK exp	A TCP ACK packet is dropped if the TCP ACK number was seen with a higher value than the FIN_ACK expected TCP ACK value.
[Drops] fp TCP exceeded MSS	A TCP Packet is dropped if the sent TCP segment's length was greater than the MSS earlier advertised (in TCP SYN options field).

[Drops] fp IP TTL is zero	An Ipv4 Packet is dropped if received with a Ipv4 TTL value of zero.
Fastpath generated TCP ack	The following statistic is the number of "ACK challenges" sent by the ACE. This occurs when the ACE receives: 1. A TCP SYN on an established connection, 2. A TCP packet with an ACK for a sequence number which is beyond the data which has been sent, or 3. A TCP RST with an ACK for a sequence number which is beyond the data which has been sent, then the ACE will DROP the incoming TCP packet and return a TCP ACK to the "sender". The sender is the station identified in the connection, and is not the station (MAC) from which we got the packet.
Note: The following are context ALL statistics.	
L3 invalid header len	An Ipv4 Packet is dropped if IP header length is less than 20 bytes or greater than the total IP length field.
L3 L2/L3 length mismatch	An Ipv4 Packet is dropped if IP total length exceeds the ethernet/IMPH total length.
L3 TTL repl	This statistic is incremented when a Ipv4 packet is received and the Ipv4 TTL was replaced with the minimum configured TTL configured on the vlan interface if lower.
L3 invalid flags	Deprecated.
L3 options removed	This statistic is incremented when all the Ipv4 options on the configured vlan interface were cleared.
L3 invalid options	This statistic is incremented when the invalid Ipv4 options on the vlan interface are cleared. This is enabled by default.
L3 DF cleared	This statistic is incremented when the Don't Fragment (DF) bit in the Ipv4 header was configured to be cleared on the vlan interface.
L3 invalid address	This statistics is incremented for invalid IP Source and Destination addresses (except on internal VLAN 1, eg. 0.0.0.0, 255.255.255.255, 127.x.x.x., etc)
[Drops] L4 invalid header len	This statistics is incremented for packets dropped when the TCP/UDP header length is less than defined by standard or is greater than TCP/UDP total packet length.
[Drops] icm TCP normalization	All Normalization is done either in Fastpath or in ICM. These are the aggregate statistics for all the normalization drops done in either of these microengines. ICM tcp normalization happens for Layer 7 (L7) Load Balanced (LB) connectoions (flows) if normalization is disabled in the fastpath. TCP Non-syn packets are sent to ICM for L7 LB connections (flows) and they get dropped by ICM normalization. One possible reason for this to happen is if a connection gets closed abruptly by a TCP RST and there are in-flight packets received by ACE after the connection record was freed
[Drops] fp TCP normalization	This is incremented because of other normalization drops such as "fp TCP ack past seq" or "fp TCP window left edge".

show np 1 me-stats -socm

Displays information for the outbound connection manager (OCM) microengine.

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-socm -v"
OCM Statistics: (Current)
--------------
Errors:                                        0            0
Connection create received:                25005            0
LB dest decision received:                 34701            0
Nat app fixup recieved:                        0            0
Connection unproxy received:                   0            0
Connection reproxy received:                   0            0
IPCP received:                                 0            0
ACK trigger received:                          0            0
TCP connected received                      9532            0
Unknown message received:                      0            0
Drop [LB dest decision fail]:                  0            0
Drop [invalid ifid]                            0            0
Drop [Out of buffers]:                         0            0
Dest decision transmitted:                  9532            0
TCP connect transmitted:                    9723            0
ACK trigger transmitted:                       0            0
IPCP transmitted:                              0            0
NAT[static mapped]:                            0            0
NAT[static real]:                              0            0
NAT[xlate alloc fail]:                         0            0
NAT[xlate real hit]:                           0            0
NAT[xlate mapped hit]:                         0            0
NAT[invalid xlate]:                            0            0
NAT[dump xlate]:                               0            0
NAT[xlate release failed]:                     0            0
NAT Pool Alloc [fail]:                         0            0
NAT Pool Alloc [addr]:                         0            0
NAT Pool Alloc [addr/port]:                    0            0
NAT Pool Free [addr]:                          0            0
NAT Pool Free [addr/port]:                     0            0
NAT Pool Free [orphan IP]:                     0            0
Reuse retrieve link update conn invalid        0            0
Reuse retrieve link update conn not on r       0            0
Reuse retrieve success but conn invalid:       0            0
Drop [Next Hop queue full]:                    0            0
Reuse retrieve miss:                           0            0
OCM Packet count (Hi & Lo):                69238            0
Packet forward received:                       0            0
UDP Chaser received:                       77664            3
NAF Error [no route or unresolved adjace       0            0
NAF Error [nat resp fail]:                     0            0
(Context ALL Statistics)
Drop [out of connections]:                     0            0
Drop [out of proxies]:                         0            0
Drop [out of ssl]:                             0            0
Drop [mac lookup fail]:                        0            0
Drop [route lookup fail]:                  25005            0
Drop [nat fail]                                0            0
Drop [ip sanity check fail]                    0            0
Drop [acl deny]:                               0            0
Drop [redundant connection]:                   0            0
Connection inserted:                       34701            0
Packet message transmitted:                    0            0
Reuse conns retrieved:                         0            0
Drop [Reproxy fail]:                           0            0
Drop [dest nat fail]:                          0            0
```

Notes

Field	Description
Errors	The outbound connection manager (OCM) activate connection operation failed.
Connection create received	Routed connection
LB dest decision received	Received destination decision from load balancer.
Nat app fixup recieved	Received the NAT app fixup request.
Connection unproxy received	Received unproxy request from the TCP engine.
Connection reproxy received	Received reproxy request from TCP or FPTX
IPCP received	IPCP for remote NAT alloc/release.
ACK trigger received	Received ACK trigger from fp
TCP connected received	TCP connection successful
Unknown message received	OCM hi/lo queue receive unknown message
Drop [LB dest decision fail]	The sum of "L4 Rejected Conns" and "L7 Rejected Conns" from -slb.
Drop [Out of buffers]	Particle allocation error when trying to send inter-IXP message for NAT allocation.
Dest decision transmitted	After outbound creation, sent the destination decision to HTTP or LB.
TCP connect transmitted	Outbound connection activate; TCP/UDP connection request sent
ACK trigger transmitted	Finish process ack trigger message and sent out to TCP_RX
IPCP transmitted	Sent inter-NP message for remote NAT allocation
NAT[static mapped]	Mapped address static NAT translate policy hit and translated.
NAT[static real]	Real address static NAT translated.
NAT[xlate alloc fail]	Dynamic NAT allocation failure.
NAT[xlate real hit]	Lookup xlate by real address success.
NAT[xlate mapped hit]	Lookup xlate by mapped address success.
NAT[invalid xlate]	Lookup xlate by mapped address return xlate entry seq mismatch.
NAT[dump xlate]	xlate entries cleared.
NAT[xlate release failed]	Release xlate failed the ref count already 0.
NAT Pool Alloc [fail]	NAT allocation for address/port failed.

NAT Pool Alloc [addr]	NAT allocation for address succeeded.
NAT Pool Alloc [addr/port]	NAT allocation for port success.
NAT Pool Free [addr]	NAT release for address succeeded.
NAT Pool Free [addr/port]	NAT release for port succeeded.
NAT Pool Free [orphan IP]	Couldn't find pool to which the IP address belongs.
Reuse retrieve link update conn invalid	Setting the reuse pool next reuse object hit invalid connection.
Reuse retrieve link update conn not on r	Setting the reuse pool; reuse object hit connection does not have reuse_fl set.
Reuse retrieve success but conn invalid	Outbound or proxy connection invalid.
Drop [Next Hop queue full]	OCM failed to post message to next queue.
Reuse retrieve miss	Couldn't find a connection in the TCP reuse list matching criteria.
Packet forward received	Packet forwarded.
UDP Chaser received	Incremented when the ocm receives the chaser message from ICM, as the result of a partial/missed connection. See **-sicm** stats for more detail.
OCM Packet count (Hi & Lo)	Check ocm received packet count for microengine (me) hang detect.
Drop [out of connections]	Connection allocation returned 0; resource policy does apply here.
Drop [out of proxies]	Could not allocate proxy entry for outbound connection creation.
Drop [out of ssl]	Unused
Drop [mac lookup fail]	MAC lookup if connection is bridged; could not find an encap ID for the destination MAC address.
Drop [route lookup fail]	No route to destination. This indicates the absence of a route or encaps for the destination. This can happen if the route is not configured properly. Probes to be sent to a remote rserver which is not accessible because the gateway is down will cause this counter to increment.
Drop [ip sanity check fail]	Post NAT IP sanity check; verify that the source and destination IP are different.
Drop [acl deny]	Outbound ACL deny.
Drop [redundant connection]	Trying to add a new connection and already found an existing one. One way this can occur is in the case of two VLANs on the client VIP side of ACE. If a client sends a message to a single destination over both VLANs, the ACE will attempt to setup the connection on each, resulting in a connection collision.

	Another situation where this counter may increase is, with both tranparent LB and Persistence rbalance enabled, the ACE sometimes RSTs the client connection (see description of CSCtc73599 in the Bug Toolkit).
Connection inserted	Connections inserted into the route table.
Packet message transmitted	Packets sent out.
Reuse conns retrieved	Get connection from reuse pool.
Drop [Reproxy fail]	Reproxy failed because there was a failure to read action node or the proxy allocation failed (resource policy does apply).
Drop [dest nat fail]	Say there is a PAT pool and a client address is translated to one of the PAT pool addresses. Now if the server, for whatever reason, initiates a new connection to the PAT pool address then the "nat fail" drop statistics is incremented. ACE does not allow connections to the PAT pool addresses (unless inspect is enabled).

show np 1 me-stats -sreass

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-sreass -v"
REASSEMBLY Statistics: (Current)
--------------
Rx Timer Event:                              5862798            10
Rx Syslog Request:                                 0             0
Rx Fragment Request:                               0             0
Rx Reass Request:                                 96             0
Rx Debug Request:                                  0             0
Rx Other:                                          0             0
Tx Fragment Frame:                                 0             0
Tx Reassmbled Frame:                               0             0
Tx To Other IXP:                                   0             0
Tx Debug Frame:                                    0             0
Tx Syslog Frame:                                   0             0
Total Datagram count:                             32             0
Timeouts:                                         24             0
DROP: Timeout Drops:                              32             0
DROP: Error Drops:                                 0             0
DROP: Overlap Drops:                               0             0
DROP: Duplicate Drops:                             0             0
DROP: Over-limit Drops:                            0             0
DROP: Unallowed Frag Drops:                        0             0
DROP: Fragment ip len below threshold:            64             0
Pending Particle Count:                            0             0
```

Notes

Field	Description
Rx Timer Event:	TCP timer Expiration/Free/Reuse/Stop message received.
Rx Syslog Request:	Count of syslog messages received.

Rx Fragment Request:	Count of requests to fragment packets according to egress MTU.
Rx Reass Request:	Count of requests to reassemble received fragments.
Rx Debug Request:	Count of packets sent to packet capture.
Rx Other:	Deprecated (received request to compute checksums).
Tx Fragment Frame	Count of packets fragmented by this IXP according to egress MTU.
Tx Reassmbled Frame:	Count of packets reassembled by this IXP and sent to Fastpath.
Tx To Other IXP:	Count of packets reassembled and punted to other IXP (includes the punt packets which are fragmented if size > 12K bytes)
Tx Debug Frame:	Count of packets sent to packet capture process.
Tx Syslog Frame:	Count of packets sent to syslog process.
Total Datagram count:	Count of packets hashed/reassembled/dropped.
Timeouts:	Count of fragment chains dropped after timeouts
DROP: Timeout Drops:	Count of fragment dropped after timeouts.
DROP: Error Drops:	Count of fragment dropped which were received with invalid interface ids.
DROP: Overlap Drops:	Count of fragment dropped which overlapped the packet payload of existing fragments.
DROP: Duplicate Drops:	Count of fragment dropped which contained a duplicate fragment of already existing fragments.
DROP: Over-limit Drops:	Count of fragment dropped when total current fragments in reassembly reaches buffer limit of 10000 buffers or when hash entry has reached high watermark of 9500 fragments or when current fragment chain has exhausted the configured fragment chain limit.
DROP: Unallowed Frag Drops:	Count of fragment dropped when interface has been configured with not having any fragment chain.
DROP: Fragment ip len below threshold:	Count of fragment dropped when length of IPv4 packet is less than configured min-mtu for reassembly.
Pending Particle Count:	Count of current number of buffers in all fragments in all fragment chains in reassembly.

show np 1 me-stats -sreceive

This command and "show np 1 me-stats -srx" are the same.

Sample Output

```
ACE30002/Admin#  show np 1 me-stats "-sreceive -v"
Receive Statistics: (Current)
------------------
Idle:                                  3011935060        100914
Frames Received:                         20228554            43
Control Frames Received:                 13525160            26
```

```
Forward RBUF:                                0            0
Forward RBUF+DRAM:                           0            0
Forward Buffered:                     20228554           43
Post stalls:                                 0            0
Packet drops:                                0            0
Error(bad rbuf):                             0            0
Error(missing eop):                          0            0
Error(missing sop):                          0            0
Last bad RBUF control word:                  0            0
Error(data buf alloc fail):                  0            0
Error(control buf alloc fail):               0            0
```

Notes

Field	Description
Idle	0x952d3a9e
Frames Received	Count of all frames received (includes low and high priority frames)
Control Frames Received	Count of high priority frames received
Forward RBUF	Deprecated stat
Forward RBUF+DRAM	Deprecated stat
Forward Buffered	Count of buffers forwarded to fastpath
Post stalls	Number of times packet forwarding to fastpath failed due to FP queue being full.
Packet drops	Number of packets dropped, due to post stall or receive error.
Error(bad rbuf)	Count of SPI (fabric) errors detected on receive. Usually recoverable.
Error(missing eop)	Count of cases where SOP (Start Of Packet) indicator was received before EOP (End Of Packet). Usually indicates a CDE problem.
Error(missing sop)	Count of cases where SPI frame following an EOP frame did not have the expected SOP indicator. Usually indicates a CDE problem.
Last bad RBUF control word	Contains last control word of the last bad receive buffer.
Error(data buf alloc fail)	Failed to get a receive buffer; packet is dropped
Error(control buf alloc fail)	Failed to get a control buffer. This is incremented once per buffer. ACE will then loop forever until one becomes available. Therefore the buffer will be allocated or ACE will crash.

show np 1 me-stats -stcp

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-stcp -v"
TCP Statistics: (Current)
--------------
TCP RX messages received:                   176605            0
TCP RX unknown messages:                         0            0
TCP RX racing messages (fin):                    0            0
```

```
TCP RX racing messages (forward):                1            0
TCP RX racing messages (conn create):            0            0
TCP TX messages received:                   163798            0
TCP TX Hi Priority messages received:        14773            0
TCP TX unknown messages:                         0            0
TCP TX racing messages (connect):                0            0
TCP TX racing messages (data):                   0            0
TCP TX racing messages (proxy):              50069            0
Reproxy message received:                        0            0
Data messages received:                      94194            0
TCP connect message received:                 9745            0
Ack trigger message received:                    0            0
Unproxy req. message received:                   0            0
Unproxy rsp. message received:                   0            0
TCP accepted msgs sent:                       9745            0
TCP connected msgs sent:                      9553            0
Conn_ctrl msgs sent:                         16120            0
Buffer alloc failed:                             0            0
Invalid msg ring id:                             0            0
Start retrans timer:                        115174            0
Start ackdelay timer:                        37544            0
Start persist timer:                             0            0
Start timewait timer:                            0            0
Delete act timer:                            37544            0
Delete rtp timer:                            99218            0
Connections unproxying:                          0            0
Connections unproxying canceled by TCP:          0            0
Connections unproxying canceled by app:          0            0
Connections unproxying immediate reproxy         0            0
Connections unproxying flush retransq:           0            0
Connections unproxying flush inputq:             0            0
Connections unproxied:                           0            0
Connections reproxied:                           0            0
Drop reproxy msg queue full:                     0            0
Drop control msg:                                0            0
Drops due to FastTX queue full:                  0            0
Drops due to Fastpath queue full:                0            0
Drops due to HTTP queue full:                    0            0
Drops due to SSL queue full:                     0            0
Drops due to AI queue full:                      0            0
Drops due to Fixup queue full:                   0            0
Drops due to packet size exceed MSS:             0            0
Unproxy rsp post failed:                         0            0
Drops due to invalid proxy id:                   0            0
Drops due to UDP buffer share limit:             0            0
(Context ALL Statistics)
Handshakes completed:                        19298            0
Handshakes failed:                             192            0
Packets received to app:                     85025            0
Packets sent to network:                    184248            0
Segs outside window:                             0            0
ACK past SEQ:                                    0            0
Dup ACKs received:                            3206            0
Dup ACK limit met:                               0            0
Malformed TCP options:                           0            0
Reassemble segs:                                 0            0
Nagled data segs:                                0            0
Retransmitted data segs:                         3            0
Round-trip timeouts:                          5028            0
Round-trip timeout limit met:                  192            0
Persist timeouts:                                0            0
Persist timeout limit met:                       0            0
Ack delay timeouts:                              0            0
Timewait timeouts:                               0            0
```

```
Connection shutdown FIN:                    18438            0
Connection shutdown RST:                      334            0
SYNs received:                              19298            0
FINs received:                              18438            0
ACKs received:                             166859            0
RSTs received:                                526            0
PSHes received:                             47289            0
SYNs transmitted:                           24323            0
FINs transmitted:                           18438            0
ACKs transmitted:                          169669            0
RSTs transmitted:                             334            0
PSHes transmitted:                          84644            0
```

Notes

Field	Description
TCP RX messages received	Total messages received by the TCPRX engine
TCP RX unknown messages	Unknown message types, expect 0 (non-fatal)
TCP RX racing messages (fin)	Received a FIN and seq is out of order (non-fatal)
TCP RX racing messages (forward)	Could be invalid proxy id or invalid seq# (non-fatal)
TCP RX racing messages (conn create)	Reproxy or conncreate and state different than listen (non-fatal)
TCP TX messages received	Total messages received by the TCPTX engine
TCP TX Hi Priority messages received	Total high priority messages received by the TCPTX engine
TCP TX unknown messages	Unknown message types, expect 0 (non-fatal)
TCP TX racing messages (data)	TCP transmit message data with invalid seq or state TCB_FREE (non-fatal)
Reproxy message received	Subsequent request received, set up a new TCB
Data messages received	Data message from application to be sent
TCP connect message received	Application has requested active connect
Ack trigger message received	Message signifies client has received full server response
Unproxy req. message received	Application is initiating unproxy operation
Unproxy rsp. message received	Application is finishing unproxy operation (may cancel or confirm)
TCP accepted msgs sent	3-way handshake completed
TCP connected msgs sent	Syn received
Conn_ctrl msgs sent	Connection ctrl msg sent, that is, embryonic connection failure or TCP receive
Buffer alloc failed	System is out of internal data buffers, no action taken
Invalid msg ring id	Application is not a valid next_hop (expect 0)
Start retrans timer	Retransmit timer started when data packet sent
Start ackdelay timer	Ackdelay timer started when packet accepted and nagle configured
Start persist timer	TCP probe; TCP wnd zero
Start timewait timer	Timer started when timewait state reached (not implemented)

Delete act timer	Ack delay timer stopped when ack sent or other misc. reasons
Delete rtp timer	Retransmit timer stopped when ack received
Connections unproxying	Only unproxy if response analysis done, no buffered data and not closing
Connections unproxying canceled by TCP	TCP moved out of estab state before unproxy finished (e.g., fin received)
Connections unproxying canceled by app	Application received data during 3 way unproxy handshake, decided to stay proxied
Connections unproxying immediate reproxy	TCP received data before retransmit queue flushed, but after application unproxied
Connections unproxying flush retransq	The application unproxied, but data still not acked by the endpoint
Connections unproxying flush inputq	The application unproxied, and TCP received data during the unproxy, which is being flushed to fastpath
Connections unproxied	Number of times TCP successfully unproxied
Connections reproxied	Number of times TCP successfully reproxied
Drop reproxy msg queue full	TCP failed to notify app of reproxy, conn will be closed
Drop control msg	TCP failed to notify app of new conn, conn will be closed
Drops due to FastTX queue full	TCP failed to send packet to network, packet is dropped
Drops due to Fastpath queue full	TCP failed to send packet to network, packet is dropped
Drops due to HTTP queue full	TCP failed to send packet to http, packet is dropped (and not acked)
Drops due to SSL queue full	TCP failed to send packet to ssl, packet is dropped (and not acked)
Drops due to AI queue full	TCP failed to send packet to appinspect, packet is dropped (and not acked)
ACK past SEQ	Received ack is past our next snd sequence number
Unproxy rsp post failed	Unused
Drops due to invalid proxy id	Message received with sequence mismatch or tcb already freed, ignore message
Handshakes completed	Accepted plus connected connections
Handshakes failed	Something failed while state not yet established; could be persist timeout, retrans timeout, or RST packet received
Packets received to app	TCP send data message counter; data or FIN sent to application
Packets sent to network	TCP transmission to fastpath
Segs outside window	A packet is dropped due to receive window check failure
Dup ACKs received	The endpoint is re-sending an ack
Dup ACK limit met	Limit is 3 duplicate acks; begin fast retransmit
Malformed TCP options	Unknown TCP option received
Reassemble segs	Reassembled TCP segments
Nagled data segs	Outgoing data delayed and collected to form 1 MSS segment
Retransmitted data segs	Number of packets sent after a retransmit timer expired
Round-trip timeouts	Retransmit timer expired
Round-trip timeout limit met	Segment transmitted 4 times, considered failure, connection reset
Persist timeouts	The persist timer expired
Ack delay timeouts	ACK delay timer expired; ACK message will be sent

Timewait timeouts	Unused
Connection shutdown FIN	Application initiated graceful connection close
Connection shutdown RST	Application initiated immediate connection close
SYNs received	SYN flag received
FINs received	FIN flag received
ACKs received	ACK flag received
RSTs received	RST flag received
PSHes received	PSH flag received
SYNs transmitted	SYN flag transmitted
FINs transmitted	FIN flag transmitted
ACKs transmitted	ACK flag transmitted
RSTs transmitted	RST flag transmitted
PSHes transmitted	PSH flag transmitted

show np 1 me-stats -stimer

Displays information on activities of the timer management ME within ACE.

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-stimer -v"
Timer Statistics: (Current)
------------------
ME Timer Expirations:                       2866588             5
ME Timer User Adds:                          152737             0
ME Timer Internal Adds:                      151416             0
ME Timer User Dels:                          146386             0
ME Timer User Frees:                         151414             0
ME Timer Freed Internal Count:                 1321             0
ME Timer Marked For Deletion:                  1321             0
ME Timer Unexpected Req Rcved:                    0             0
ME Timer User Del Aborted:                        0             0
ME Max Timers Used:                               5             0
ME Timer Freelist Empty:                          0             0
ME Timer Reused:                            2861559             4
```

Notes

Field	Description
ME Timer User Adds	Timer values from these timers: timer from: nat_release_dynamic_xlate, frag_init, ssl_me_transmit_data_packet, tcp_rx_reassemble, tcp_rx_timewait, tcp_retrans_expire, tcp_persist_expire, tcp_timewait_expire
ME Timer Internal Adds	The user add has been processed
ME Timer User Dels	Mark timer as deleted by application
	Application released the timer ID, returned to free list

ME Timer User Frees	
ME Timer Freed Internal Count:	Application free request processed internally
ME Timer Marked For Deletion:	Application requested delete of already freed timer (non-fatal)
ME Timer Unexpected Req Rcved:	Unknown message received by timer ME, ignored
ME Timer User Del Aborted	User requested timer stop before expiration
ME Max Timers Used	The peak timer usage
ME Timer Freelist Empty	Add requested but no timers were available
ME Timer Reused:	Expired timer is restarted (reassembly does this periodically)

show np 1 me-stats -t

Displays the valid TCP proxy table entries (TCBs). There are two ways to view/display the TCB associated to the connection:

1. Knowing the proxy ID of the connection from the "Proxy ConnID" field from "sh np 1 me-stats "-c <id>", and issuing ""sh np 1 me-stats "-t <proxy_id>"".
2. Or using the "-t" switch along with "-c", such as "sh np 1 me-stats "-c <cid> -t""

Given this, for L4 connections, ACE won't maintain any TCB, so the proxy id will be also zero. But if the "sh np 1 me-stats "-c <cid> -t"" issued for L4 connection, then the connection id is treated as proxy ID, and displays the stale values available at that memory location.

Sample Output

```
ACE30002/Admin# show np 1 me-stats "-t -v"
TCB for ProxyID.seq: 1.13[0x1.0xd]
------------------------------
State: 3                Flags: 0x80800001
PATH-ID: 0x1            Context-ID: 0x2
snd_una: 0xfbcd6345             snd_cur: 0xfbcd6345
snd_wl1: 0xb694e255
snd_nxt: 0xfbcd6345             snd_mss: 0x5b4
snd_wnd: 0x16d0         proxyrtt: 0x0   cwnd: 0xffffff
rcv_nxt: 0xb694e299     rcv_wnd: 0x7fbc
recover: 0x0            bufshr config: 0x40
rto    : 0xbb8          rtt_start: 0x2e957742
rtt_seq: 0x0            srtt: 0xbb8             rtt_var : 0x1
ssthresh: 0xffff        retranslen: 0x0         retransq : 0x0
act_timer: 0x0          rtp_timer: 0x0
rcv_mss: 0x5b4          WindowScale rx/tx: 0x0/0x0
inputq: 0x0             ackdelto: 0xc8
maxtries: 0x4           tries: 0x0
dup_ack: 0x0            reassembleq: 0x0
```

```
iss: 0xfbcd6344          irs: 0xb694e254
ts_recent: 0x0           ts_last_ack: 0x0
ts_last_sent: 0x0                swsthresh: 0x0
ts_delta: 0x0            finooo_reasslen: 0x0
Buffer Share Usage: 512

TCB for ProxyID.seq: 4.9[0x4.0x9]
------------------------------
State: 1                 Flags: 0x80001002
PATH-ID: 0x2             Context-ID: 0x2
snd_una: 0xe628c699              snd_cur: 0xe628c69a
snd_wl1: 0x0
snd_nxt: 0xe628c69a              snd_mss: 0x5b4
snd_wnd: 0x0             proxyrtt: 0x0   cwnd: 0x5b4
rcv_nxt: 0x0    rcv_wnd: 0x8000
recover: 0x0             bufshr config: 0x40
rto    : 0x1770          rtt_start: 0x0
rtt_seq: 0x0             srtt: 0xbb8             rtt_var : 0x1
ssthresh: 0xb68 retranslen: 0x0          retransq : 0x0
act_timer: 0x0           rtp_timer: 0x3ad90488
rcv_mss: 0x5b4           WindowScale rx/tx: 0x0/0x0
inputq: 0x0              ackdelto: 0xc8
maxtries: 0x4            tries: 0x2
dup_ack: 0x0             reassembleq: 0x0
iss: 0xe628c699          irs: 0x0
ts_recent: 0x0           ts_last_ack: 0x0
ts_last_sent: 0x0                swsthresh: 0x0
ts_delta: 0x0            finooo_reasslen: 0x0
Buffer Share Usage: 0
```

Passing a connection ID, with the -c yields information for that connection. The following sample shows the output for a particular connection by ID discovered with the show connection command.

```
switch/Admin# sh conn protocol tcp

conn-id    np dir proto vlan source                destination           state
----------+--+---+-----+----+---------------------+---------------------+------+
1          1  in  TCP   40   209.165.201.1:56873     209.165.201.21:80      ESTAB
2          1  out TCP   50   209.165.202.129:80      209.165.201.1:56873       ESTAB

switch/Admin# sh np 1 me-stats "-c 1 -t"
TCB for ProxyID.seq: 1.0[0x1.0x0]       <<<<<
------------------------------
State: 0                 Flags: 0x0
PATH-ID: 0x0             Context-ID: 0x0
snd_una: 0x0             snd_cur: 0x0
snd_wl1: 0x0
snd_nxt: 0x0             snd_mss: 0x0
snd_wnd: 0x0             proxyrtt: 0x0   cwnd: 0x0
rcv_nxt: 0x0    rcv_wnd: 0x0
recover: 0x0             bufshr config: 0x0
rto    : 0x0             rtt_start: 0x0
rtt_seq: 0x0             srtt: 0x0               rtt_var : 0x0
ssthresh: 0x0   retranslen: 0x0          retransq : 0x0
act_timer: 0x0           rtp_timer: 0x0
rcv_mss: 0x0             WindowScale rx/tx: 0x0/0x0
inputq: 0x0              ackdelto: 0x0
maxtries: 0x0            tries: 0x0
dup_ack: 0x0             reassembleq: 0x0
iss: 0x0                 irs: 0x0
ts_recent: 0x0           ts_last_ack: 0x0
ts_last_sent: 0x0                saved_qaddr: 0x0
ts_delta: 0x0            finooo_reasslen: 0x0
```

```
Buffer Share Usage: 0

Connection ID:seq: 1[0x1].0     <<<<<
  Other ConnID    : 2[0x2].0
  Proxy ConnID    : 0[0x0].0    <<<<<
  Next Q    : 0[0x0]

209.165.201.1:56873 -> 209.165.201.21:80 [RX-NextHop: TX] [TX-NextHop: TX]
  Flags:  PAT: No  DynNAT: No  Implicit PAT: No On_Reuse: No
  L3 Protocol     : IPv4                L4 Protocol    : 6
  Inbound Flag    : 1
  Interface Match : Yes
    Interface MatchID: 3
  EncapsID:ver    : 7:0         TCP ACK delta  : 0x69f5cc14
  MSS             : 1460                TOS Stamp       : 0
  Repeat mode     : No          ARP Lookup      : No
  TOS Stamp       : No          TCP Window Check: No
  ACE ID          : 167         NAT Policy ID       : 0
  Post NAT hop    : 0
  Packet Count    : 2           Byte Count          : 100
  TCP Information: (State = 3)
    Window size   : 5840                Window scale    : 0
    FIN seen      : No          FIN/ACK seen    : No
    FIN/ACK exp   : No          Close initiator : No
    FIN/ACK expval: 5b40000             Last seq        : dae90377
   timestamp_delta: 0           Last ack        : 6ec8aa39
    No Trigger    : 0           Trigger Status   : 0
    Timestamp : 5f322
  TCP options negotiated:
    Sack:Clear          TS:Clear        Windowscale:  Clear
    Reserved: Allow     Exceed MSS:  Deny       Window var: Allow

Raw Connection Entry
0000  0x00000000  0x280a0001  0x280a6479  0x06090003
0010  0xde290050  0x69f5cc14  0x00070000  0x05b40000
0020  0x00000002  0x00000000  0x02080480  0x24450027
0030  0x00000002  0x00000064  0x16d00030  0x05b40000
0040  0xdae90377  0x6ec8aa39  0x0005f322  0x00000000
0050  0x000000a7  0x00000000  0x0005e2be  0x00000000
0060  0x00000000  0x00000000  0x00000000  0x00000000
switch/Admin#
```

Notes

In the above ouptut, the "Proxy ConnID" is zero. But still the "-t" option treated the "Connection ID" as "Proxy ConnID" and displayed the stale details available in that memory location.

show np 1 me-stats -u

Displays ME ring utilization. This is done by monitoring how frequently each process services its receive queue. The more the receive queue is serviced, the less busy the process is presumed to be. Therefore, since a process that is hung or has crashed cannot service its queue, it is shown as having 100% ME utilization.

Sample Output

```
ACE30002/Admin# show np 1 me-stats -u
ME Utilization Statistics
--------------
```

```
RECEIVE:                                        0
FASTPATH:                                       0
SLOWTX:                                         0
TCP_RX:                                         0
HTTP:                                           0
IH_RX                                           0
SSL_ME:                                         2
CM_CLOSE:                                       0
X_TO_ME:                                        0
FIXUP:                                          0
REASSEMBLY:                                     0
OCM:                                            0
TCP_TX:                                         0
ICM:                                            0
```

Notes

Related command is show system resources (show overall CPU).

show np 1 me-stats -x

Displays the valid xlate (IP address and port translation) entries. This command only shows details for 10 entries and then gives the total of the number of active xlates per NP.

Sample Output

```
ACE30002/Admin#  show np 1 me-stats -t
Total currently active TCBs = 0

ACE30002/Admin# show np 1 me-stats -x
xlate id: 0x1[seq=4]
policy id: 2
real_ifid: 6            Real address: 172.19.133.169
mapped_ifid: 5          Mapped address: 209.165.202.129
next real: 0x0          next mapped: 0x0
timeout: 0x2a30 timestamp: 0xee4e058
pool_id: 2              ref_cnt: 2
state: dump
[snip]
policy id: 4
real_ifid: 6            Real address: 172.19.133.169
mapped_ifid: 5          Mapped address: 209.165.202.129
next real: 0x40000001           next mapped: 0x40000001
timeout: 0x2a30 timestamp: 0x16225569
pool_id: 5              ref_cnt: 0
timer: 0x0              state: valid
4 active xlate(s).
```

Notes

Use "show xlate | [options]" to get more information on xlates. The options are:

```
ACE30002/Admin# show xlate ?
  global  Show current translation by Global address
  gport   Show current translation by Global port
  local   Show current translation by local address
  lport   Show current translation by local port
```

To see the number of XLATES per NP:

```
ACE30002/Admin# show np 1 me-stats -x

xlate id: 0xa[seq=1]
policy id: 7
real_ifid: 74           Real address: 10.40.225.166
mapped_ifid: 45         Mapped address: 209.165.202.129
next real: 0x10007baa           next mapped: 0x0
timeout: 0x4b0  timestamp: 0x448d5034
pool_id: 3              ref_cnt: 0
timer: 0x0              state: valid
65532 active xlate(s).
```

show np 1 me-stats -y

Displays system information gathered during device bootup. This information remains unchanged while the device is running.

In particular, this command is used to gather microengine allocation information. ME allocation is useful when you need to figure out which ME cored from the ME number in the core file name.

Sample Output

```
ace4/Admin# show np 1 me-stats -y
Microcode Built at [12/10/07_@_17:25] by [adbuild]
Product ID/Rev: 17  Clock Speed: 1400Mhz   Strap Options: 5c00
DRAM 0: Size: 1536MB   Available: 128MB   Offset: 0
   (Older hw revs had size < 1536 MB)

SRAM 0: Size: 8MB   Offset: 0
SRAM 1: Size: 8MB   Offset: 0
SRAM 2: Size: 8MB   Offset: 0
SRAM 3: Size: 8MB   Offset: 0

Microengine Allocation:
RX:              0
Fastpath:        1, 2, 21, 22, 23
ICM:             3
App-Fixup:       4
TCP-RX:          5
TCP-TX:          6
Timers/Reass:    7
OCM:             16
CM-CLOSE:        17
HTTP:            18, 19
SSL-ME:          20
ace4/Admin#
```

Notes

There are sixteen microengines on each IXP, which are numbered 0-7 and 16-23. Here is the ME numbering correlation:

Allocated ME	Corefile ME
0	0

1	1
2	2
3	3
4	4
5	5
6	6
7	7
16	8
17	9
18	10
19	11
20	12
21	13
22	14
23	15

show np 1 memory

Displays information on QNX processes and libraries. If a number of instances of a process are stuck (SIGWAITINFO or CONDVAR), it could point to a potential problem.

Sample Output

```
ACE30002/Admin# show np 1 memory
     pid tid name               prio STATE           code  data         stack
       1   1 proc/boot/procnto    0f READY             12K   12K      0(576)*
       1   2 proc/boot/procnto  255r RECEIVE           12K   12K      0(8192)
       1   3 proc/boot/procnto  255r RECEIVE           12K   12K      0(8192)
       1   4 proc/boot/procnto   10r RECEIVE           12K   12K      0(8192)
       1   5 proc/boot/procnto   10r RECEIVE           12K   12K      0(8192)
       1   6 proc/boot/procnto   10r RECEIVE           12K   12K      0(8192)
       1   7 proc/boot/procnto   10r RUNNING           12K   12K      0(8192)
       1   8 proc/boot/procnto   11r RECEIVE           12K   12K      0(8192)
       1   9 proc/boot/procnto   10r RECEIVE           12K   12K      0(8192)
            procnto            @fe082000             12M
       2   1 vc-ser8250-ixp2400  10r RECEIVE           36K   56K  4096(132K)*
            ldqnx.so.2         @ 1000000             388K   20K
            /dev/mem           @ 1800000 ( 7df1000)        100K
            /dev/mem           @ 1819000 (c0030000)        4096
       3   1 proc/boot/devf-ram  10r SIGWAITINFO      104K   52K  4096(132K)*
       3   2 proc/boot/devf-ram  10r RECEIVE          104K   52K  4096(132K)
       3   3 proc/boot/devf-ram  10r RECEIVE          104K   52K  4096(132K)
            ldqnx.so.2         @ 1000000             388K   20K
            /dev/mem           @ 1800000 (       0)       8192K
  114692   1 proc/boot/devc-pty  10r RECEIVE           36K   84K  4096(132K)*
            ldqnx.so.2         @ 1000000             388K   20K
  114693   1 proc/boot/io-net    10r SIGWAITINFO       72K  328K  8192(132K)*
  114693   2 proc/boot/io-net    20r RECEIVE           72K  328K  4096(132K)
  114693   3 proc/boot/io-net    10r RECEIVE           72K  328K   8192(68K)
  114693   4 proc/boot/io-net    10r RECEIVE           72K  328K   4096(68K)
  114693   5 proc/boot/io-net    10r RECEIVE           72K  328K   4096(68K)
  114693   6 proc/boot/io-net    10r CONDVAR           72K  328K  4096(132K)
```

```
114693   8 proc/boot/io-net    10r RECEIVE          72K  328K   8192(68K)
          ldqnx.so.2         @ 1000000            388K   20K
          npm-tcpip.so       @ 1066000            764K  104K
          npm-pppmgr.so      @ 113f000             88K  8192
          devn-vr.so         @ 1157000            8192  4096
          libipcp_g.so.1     @ 115a000             16K  100K
          libsyslib_g.so.1   @ 1177000             44K  4096
          mem                @82d00000             12K  4096
          mem                @8a300000            8192  129M
          mem                @a8100000           8192K 8192K
          mem                @a9100000           8192K 8192K
          mem                @aa100000           8192K 8192K
          mem                @ab100000           8192K 8192K
          mem                @ac100000           8192K 8192K
          mem                @ad100000           8192K 8192K
          mem                @ae100000           8192K 8192K
          mem                @af100000           8192K 8192K
          mem                @b0100000            4096  4096
          mem                @b0300000            4096  4096
          mem                @b0500000            4096  4096
          mem                @b0700000            4096  4096
          mem                @b0900000             16K   16K
          mem                @b0b00000             16K
114694   1 proc/boot/sh        10r REPLY           168K   40K  4096(132K)*
          ldqnx.so.2         @ 1000000            388K   20K
114695   2 proc/boot/pipe      10r RECEIVE          16K   36K  4096(132K)
114695   3 proc/boot/pipe      10r RECEIVE          16K   36K  4096(132K)
          ldqnx.so.2         @ 1000000            388K   20K
114698   1 proc/boot/inetd     10r SIGWAITINFO      40K   40K    8192(9K)*
          ldqnx.so.2         @ 1000000            388K   20K
          libsocket.so.2     @ 1066000            128K   28K
114699   1 proc/boot/WBSrvr    10r SIGWAITINFO     296K  648K  8192(132K)*
          ldqnx.so.2         @ 1000000            388K   20K
          librpc.so.2        @ 1066000             92K  8192
          libsocket.so.2     @ 107f000            128K   28K
118792   1 proc/boot/halMeDrv  10r RECEIVE          52K   40K  4096(132K)*
          ldqnx.so.2         @ 1000000            388K   20K
          mem                @80000000             64K  4096
          mem                @80100000             64K   64K
          mem                @80300000            4096   16K
          mem                @80500000            4096  4096
          mem                @80700000            4096  4096
          mem                @80900000           8192K 8192K
          mem                @81900000           8192K 8192K
          mem                @82900000             32K   32K
          mem                @82b00000            8192  4096
118796   1 c/boot/sysmgr_g_ns  10r NANOSLEEP        8192   36K  8192(132K)*
118796   2 c/boot/sysmgr_g_ns  11r INTR             8192   36K  4096(132K)
          ldqnx.so.2         @ 1000000            388K   20K
          libsyslib_g.so.1   @ 1066000             44K  4096
          /dev/mem           @ 1800000 (d6000000)       4096
          /dev/mem           @ 1801000 ( 7df1000)       4096
          /dev/mem           @ 1802000 (c0020000)       4096
118798   1 proc/boot/rpcbind   10r SIGWAITINFO       32K  424K    8192(9K)*
          ldqnx.so.2         @ 1000000            388K   20K
          librpc.so.2        @ 1066000             92K  8192
          libsocket.so.2     @ 107f000            128K   28K
143369   1 roc/boot/ipcp_g_ns  10r RECEIVE          40K  652K  8192(132K)*
143369   2 roc/boot/ipcp_g_ns  10r NANOSLEEP        40K  652K  4096(128K)
143369   3 roc/boot/ipcp_g_ns  10r NANOSLEEP        40K  652K  4096(128K)
143369   4 roc/boot/ipcp_g_ns  10r INTR             40K  652K  128K(128K)
          ldqnx.so.2         @ 1000000            388K   20K
          libsyslib_g.so.1   @ 1066000             44K  4096
          mem                @82d00000             12K 1040K
```

```
            mem                 @82e00000            5264K  4096
            mem                 @83400000             512K  4096
...
            mem                 @b0c00000              50M
  147469   1 oc/boot/ha_hb_g_ns  10r JOIN                24K   36K  8192(132K)*
  147469   2 oc/boot/ha_hb_g_ns  10r CONDVAR             24K   36K  8192(132K)
  147469   3 oc/boot/ha_hb_g_ns  60r RECEIVE             24K   36K  4096(132K)
            ldqnx.so.2          @ 1000000             388K   20K
            libsyslib_g.so.1    @ 1066000              44K  4096
            libipcp_g.so.1      @ 1072000              16K  100K
            ha_user_lib_g.so.1  @ 108f000              8192  4096
            bsyslog_lib_g.so.1  @ 1092000              8192  4096
            mem                 @82d00000              12K  4096
            mem                 @8a100000             4096  4096
            mem                 @8a300000             8192 1024K
...
            mem                 @b0b00000              16K
  151567   1 c/boot/sdwrap_g_ns  10r CONDVAR            8192   36K  8192(132K)*
            ldqnx.so.2          @ 1000000             388K   20K
            libsyslib_g.so.1    @ 1066000              44K  4096
            libipcp_g.so.1      @ 1072000              16K  100K
            mem                 @82d00000              12K  4096
            mem                 @8a000000            1024K  8192
...
            mem                 @b0a00000              16K   16K
  155664   1 boot/setClock_g_ns  10r CONDVAR            4096   36K  8192(132K)*
            ldqnx.so.2          @ 1000000             388K   20K
            libsyslib_g.so.1    @ 1066000              44K  4096
            libipcp_g.so.1      @ 1072000              16K  100K
            bsyslog_lib_g.so.1  @ 108f000              8192  4096
            code_fabric_g.so.1  @ 1092000              60K   12K
            bsdwrap_lib_g.so.1  @ 10a4000              4096  8192
            mem                 @82d00000              12K  4096
            mem                 @8a000000            1024K  8192
...
            mem                 @b0a00000              16K   16K
  159761   1 oot/dumper_cp_g_ns  10r RECEIVE            72K   48K  8192(132K)*
            ldqnx.so.2          @ 1000000             388K   20K
            libsyslib_g.so.1    @ 1066000              44K  4096
            libipcp_g.so.1      @ 1072000              16K  100K
            bsyslog_lib_g.so.1  @ 108f000              8192  4096
            code_fabric_g.so.1  @ 1092000              60K   12K
            mem                 @82d00000              12K  4096
            mem                 @8a300000             8192  4096
            mem                 @a0000000             129M  4096
...
            mem                 @b0b00000              16K
  163858   1 /showProcInfo_g_ns  10r REPLY              32K   40K  8192(132K)*
            ldqnx.so.2          @ 1000000             388K   20K
            libsyslib_g.so.1    @ 1066000              44K  4096
            libipcp_g.so.1      @ 1072000              16K  100K
            mem                 @82d00000              12K  4096
            mem                 @8a300000             8192  129M
...
            mem                 @b0b00000              16K
  167955   1 t/loadBalance_g_ns  10r NANOSLEEP         380K  224K  8192(132K)*
  167955   2 t/loadBalance_g_ns  10r RECEIVE           380K  224K  4096(132K)
  167955   3 t/loadBalance_g_ns  10r NANOSLEEP         380K  224K  4096(132K)
  167955   4 t/loadBalance_g_ns  10r RECEIVE           380K  224K  8192(132K)
  167955   5 t/loadBalance_g_ns  10r CONDVAR           380K  224K  8192(132K)
  167955   6 t/loadBalance_g_ns  10s RECEIVE           380K  224K  4096(132K)
  167955   7 t/loadBalance_g_ns  10s RECEIVE           380K  224K  4096(132K)
  167955   8 t/loadBalance_g_ns  10s RECEIVE           380K  224K  4096(132K)
  167955   9 t/loadBalance_g_ns  10s RECEIVE           380K  224K  4096(132K)
```

```
  167955  10 t/loadBalance_g_ns  10s RECEIVE          380K  224K  4096(132K)
  167955  11 t/loadBalance_g_ns  10r NANOSLEEP        380K  224K  4096(132K)
  167955  12 t/loadBalance_g_ns  10r RECEIVE          380K  224K  4096(132K)
  167955  13 t/loadBalance_g_ns  10r NANOSLEEP        380K  224K  4096(132K)
  167955  14 t/loadBalance_g_ns  10r RECEIVE          380K  224K  4096(132K)
  167955  15 t/loadBalance_g_ns  10r RECEIVE          380K  224K  4096(132K)
            mem                @82d00000             12K 1024K
            mem                @82e00000           5264K  4096
            mem                @83900000             98M  128K
...
            mem                @d6f00000             64K
  172052   1 t/inspectHttp_g_ns  10r NANOSLEEP        228K  152K  8192(132K)*
  172052   2 t/inspectHttp_g_ns  10r NANOSLEEP        228K  152K  8192(132K)
  172052   3 t/inspectHttp_g_ns  10r CONDVAR          228K  152K  4096(132K)
            mem                @82d00000             12K 1024K
            mem                @89b00000           2880K  4096
            mem                @89e00000            416K 1024K
...
            mem                @b0c00000             50M  448M
  176149   1 ot/appInspect_g_ns  10r SEM              456K   11M  8192(132K)*
  176149   2 ot/appInspect_g_ns  10r NANOSLEEP        456K   11M  4096(132K)
  176149   3 ot/appInspect_g_ns  10r RECEIVE          456K   11M  4096(132K)
  176149   4 ot/appInspect_g_ns  10r CONDVAR          456K   11M  4096(132K)
            mem                @82d00000             12K  4096
            mem                @8a000000           1024K 1024K
            mem                @8a300000            8192  132K
...
            mem                @b3f00000            448M  128K
  180246   1 oc/boot/sslHs_g_ns  10r NANOSLEEP       1264K  228K   12K(132K)*
  180246   2 oc/boot/sslHs_g_ns  10r CONDVAR         1264K  228K   12K(132K)
  180246   3 oc/boot/sslHs_g_ns  10r CONDVAR         1264K  228K   12K(132K)
            mem                @82d00000             12K  4096
            mem                @8a000000           1024K 1024K
            mem                @8a100000            4096  4096
 ...
            mem                @d6e00000             64K
  184343   1 /boot/me_dump_g_ns  10r SIGWAITINFO      136K   44K  8192(132K)*
            ldqnx.so.2         @ 1000000            388K   20K
            libsyslib_g.so.1   @ 1066000             44K  4096
            bucdump_lib_g.so.1 @ 1072000            200K   16K
            libsme_g.so.1      @ 10a8000            8192  4096
            ucode_stats_g.so.1 @ 10ab000             44K   12K
            ap_user_lib_g.so.1 @ 10b9000             32K  4096
            libipcp_g.so.1     @ 10c2000             16K  100K
            code_fabric_g.so.1 @ 10df000             60K   12K
            libsocket.so.2     @ 10f1000            128K   28K
            mem                @9c300000            4096  4096
            mem                @a0000000            129M  4096
            mem                @a8100000           8192K   20K

...

            mem                @d6d00000            448K
 2002968   1 proc/boot/sh        10r SIGSUSPEND      168K   40K  8192(132K)*
            ldqnx.so.2         @ 1000000            388K   20K
 2002969   1 sbin/pidin          10r REPLY            44K   48K  4096(132K)*
            ldqnx.so.2         @ 1000000            388K   20K
```

Notes

While many processes displayed are QNX processes and are not very useful for debugging purposes, several are. Some of the useful processes to investigate are listed here:

Process Name	What it is
procnto	The kernel
devf-ram	Ramdisk-related QNX driver.
io-net	Driver from QNX
sh	The shell
pipe	Pipe process; only used during debugging.
WBSrvr	WebServer; not used
halMeDrv	Intel's driver, typically used only during debug and booting up.
sysmgr_g_ns	Process that monitors interrupts and watchdog.
ipcp_g_ns	IPCP driver
ha_hb_g_ns	HA heartbeat module.
sdwrap_g_ns	Debug messaging module
setClock_g_ns	Clock synchronization module
dumper_cp_g_ns	Module to collect core files. Executed only after a crash.
showProcInfo_g_ns	The module that handles show commands from the CP.
loadBalance_g_ns	The LB module.
inspectHttp_g_ns	The HTTP inspect module.
appInspect_g_ns	The app inspect module.
sslHs_g_ns	The SSL module.
me_dump_g_ns	The module that monitors ME health and utilization.

Possible thread states:

States	Description
CONDVAR	The thread is blocked on a condition variable.
DEAD	The thread has terminated and is waiting for a join by another thread.
INTERRUPT	The thread is blocked waiting for an interrupt.
JOIN	The thread is blocked waiting to join another thread.
MUTEX	The thread is blocked on a mutual exclusion lock.
NANOSLEEP	The thread is sleeping for a short time interval.
NET_REPLY	The thread is waiting for a reply to be delivered across the network.
NET_SEND	The thread is waiting for a pulse or signal to be delivered across the network.
READY	The thread is waiting to be executed while the processor executes another thread of equal or higher priority.
RECEIVE	The thread is blocked on a message receive.
REPLY	The thread is blocked on a message reply.
RUNNING	The thread is being executed by a processor.
SEM	The thread is waiting for a semaphore to be posted.
SEND	The thread is blocked on a message send.
SIGSUSPEND	The thread is blocked waiting for a signal.

SIGWAITINFO	The thread is blocked waiting for a signal.
STACK	The thread is waiting for the virtual address space to be allocated for the thread's stack.
STOPPED	The thread is blocked waiting for a SIGCONT signal.
WAITCTX	The thread is waiting for a noninteger (e.g. floating point) context to become available for use.
WAITPAGE	The thread is waiting for physical memory to be allocated for a virtual address.
WAITTHREAD	The thread is waiting for a child thread to finish creating itself.

These letters indicate the scheduling algorithm used:

- f ? FIFO scheduling
- r ? round-robin scheduling
- o ? other (currently, same as round-robin scheduling)
- s ? sporadic scheduling

show np 1 nat policies

Displays bitmaps that show currently allocated PAT ports and available ports. This can be useful when troubleshooting PAT allocation failures.

Sample Output

```
ACE30002/Admin# show np 1 nat policies

Nat tables in IXP-0:

Hash Bucket: 0
        ID:2 mapped_if:1 policy_id:1 ixp_hint:in all IXPs type:DYNAMIC nat_pool_id:2
                ID:2 PAT:1 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170
Bitmap-ID:33
                Level 1 Bitmap: 0x4c1ffff
                Level 2 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffff0000
                Level 3 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
...
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffffffe
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
Hash Bucket: 4
        ID:6 mapped_if:1 policy_id:5 ixp_hint:in all IXPs type:DYNAMIC nat_pool_id:2
                ID:2 PAT:1 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170
Bitmap-ID:33
                Level 1 Bitmap: 0x4c1ffff
```

```
                Level 2 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffff0000
                Level 3 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
...
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffffffe
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
        ID:8 mapped_if:3 policy_id:7 ixp_hint:in IXP0 type:DYNAMIC nat_pool_id:3
                ID:3 PAT:0 ixp_binding:in IXP0
                lower:172.16.183.33 upper:172.16.183.45 Bitmap:0x1ffe
Hash Bucket: 5
        ID:5 mapped_if:1 policy_id:4 ixp_hint:in all IXPs type:DYNAMIC nat_pool_id:2
                ID:2 PAT:1 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170
Bitmap-ID:33
                Level 1 Bitmap: 0x4c1ffff
                Level 2 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffff0000
                Level 3 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
...
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffffffe
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
Hash Bucket: 7
        ID:7 mapped_if:1 policy_id:6 ixp_hint:in all IXPs type:DYNAMIC nat_pool_id:2
                ID:2 PAT:1 ixp_binding:in all IXPs
                lower:172.16.182.170 upper:172.16.182.170
Bitmap-ID:33
                Level 1 Bitmap: 0x4c1ffff
                Level 2 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xfffffffffffff0000
                Level 3 Bitmap:
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff 0xffffffffffffffff
...
 0x0000000000000000 0x0000000000000000 0x0000000000000000 0x0000000000000000
```

Notes

Note that:

- 3rd level bitmap ? The 3rd level bitmap has 65536 bits each representing a port number. When a port is free the corresponding bit is set.
- 2nd level bitmap ? The 2nd level bitmap has 1024 bits, each bit corresponding to a range of 64 bits in the 3rd level bitmap. This bit is set when one or more of the corresponding 64 bits in the 3rd level bitmap is set, otherwise it is clear. The first level contains 32 bits, each bit corresponding to 32 bits in the second level.
- The mapped_if is from the output of "show interface internal iftable".
- The policy_id entry is created when a NAT action is configured in a Policy Map Class
- NAT pools are populated only in one IXP whereas PAT pools are populated in both IXPs.

show np 1 reg

This command shows register offset, register name, and register value for the the IXP2800 Media and switch fabric registers.

Sample Output

```
ACE30002/Admin# show np 1 reg
IXP 2 registers :
[0x0000]                    IXP2800_MSF_RX_CONTROL        0x40002004
[0x0004]                    IXP2800_MSF_TX_CONTROL        0x204005a8
[0x0008]             IXP2800_MSF_INTERRUPT_STATUS         0x00000000
[0x000c]             IXP2800_MSF_INTERRUPT_ENABLE         0x00000000
[0x0024]                      IXP2800_HWM_CONTROL         0x00000030
[0x002c]                IXP2800_MSF_CLOCK_CONTROL         0x00003fff
[0x0048]              IXP2800_RX_CALENDAR_LENGTH          0x00000014
[0x0050]     IXP2800_RX_THREAD_FREELIST_TIMEOUT_0         0x00000096
[0x0060]                    IXP2800_TX_SEQUENCE_0         0x800000bb
[0x0070]               IXP2800_TX_CALENDAR_LENGTH         0x00000014
[0x00a0]                      IXP2800_TRAIN_DATA         0x00000006
[0x00a4]                    IXP2800_TRAIN_CALENDAR       0x00000000
[0x0380]        IXP2800_TX_MULTIPLE_PORT_STATUS_0        0xfffcfcc0
[0x0384]        IXP2800_TX_MULTIPLE_PORT_STATUS_1        0xffffffff
[0x0500]        IXP2800_RX_PORT_CALENDAR_STATUS_0        0x00000000
[0x0504]        IXP2800_RX_PORT_CALENDAR_STATUS_1        0x00000000
[0x0508]        IXP2800_RX_PORT_CALENDAR_STATUS_2        0x00000000
[0x050c]        IXP2800_RX_PORT_CALENDAR_STATUS_3        0x00000000
[0x0510]        IXP2800_RX_PORT_CALENDAR_STATUS_4        0x00000000
[0x0514]        IXP2800_RX_PORT_CALENDAR_STATUS_5        0x00000000
[0x0518]        IXP2800_RX_PORT_CALENDAR_STATUS_6        0x00000000
[0x051c]        IXP2800_RX_PORT_CALENDAR_STATUS_7        0x00000000
[0x1000]                     IXP2800_TX_CALENDAR_0       0x00000000
[0x1004]                     IXP2800_TX_CALENDAR_1       0x00000000
[0x1008]                     IXP2800_TX_CALENDAR_2       0x00000000
[0x100c]                     IXP2800_TX_CALENDAR_3       0x00000000
[0x1010]                     IXP2800_TX_CALENDAR_4       0x00000000
[0x1014]                     IXP2800_TX_CALENDAR_5       0x00000000
[0x1018]                     IXP2800_TX_CALENDAR_6       0x00000000
[0x101c]                     IXP2800_TX_CALENDAR_7       0x00000000
```

Notes

The items displayed are documented in detail in Section 5.7 of Intel® IXP2400/IXP2800 Network Processor Programmer's Reference Manual, January 2003.