This article describes how to troubleshoot basic optimization.

Contents

Basic WAAS optimizations include TCP flow optimization (TFO), data redundancy elimination (DRE), and persistent Lempel-Ziv (LZ) compression.

TFO Troubleshooting

The number of TCP connections, their status, and disposition can give an indication of the health of the WAAS system in a specific location. A healthy system will show a large number of connections, with a significantly large percentage of these closed normally. The **show statistics tfo detail** command provides an indication of the volume, status, and disposition of connections between a particular WAAS device and other

devices in the network.

You can view global TFO statistics by using the **show statistics tfo detail** command as follows:

```
WAE# show statistics tfo detail
 Total number of connections                          : 2852
 No. of active connections                            : 3              <-----Active connections
 No. of pending (to be accepted) connections          : 0
 No. of bypass connections                            : 711
 No. of normal closed conns                           : 2702
 No. of reset connections                             : 147
    Socket write failure                              : 0
    Socket read failure                               : 0
    WAN socket close while waiting to write           : 0
    AO socket close while waiting to write            : 2
    WAN socket error close while waiting to read      : 0
    AO socket error close while waiting to read       : 64
    DRE decode failure                                : 0
    DRE encode failure                                : 0
    Connection init failure                           : 0
    WAN socket unexpected close while waiting to read : 32
    Exceeded maximum number of supported connections  : 0
    Buffer allocation or manipulation failed          : 0
    Peer received reset from end host                 : 49
    DRE connection state out of sync                  : 0
    Memory allocation failed for buffer heads         : 0
    Unoptimized packet received on optimized side     : 0
 Data buffer usages:
    Used size:          0 B,  B-size:        0 B,  B-num: 0
    Cloned size:        0 B,  B-size:        0 B,  B-num: 0
 Buffer Control:
    Encode size:        0 B,  slow:          0,  stop:          0
    Decode size:        0 B,  slow:          0,  stop:          0
 Scheduler:
    Queue Size: IO:          0,  Semi-IO:          0, Non-IO:          0
    Total Jobs: IO:    1151608,  Semi-IO:    5511278, Non-IO:    3690931

  Policy Engine Statistics
  ------------------------
  Session timeouts: 0,  Total timeouts: 0
  Last keepalive received 00.5 Secs ago
  Last registration occurred 15:00:17:46.0 Days:Hours:Mins:Secs ago
  Hits:                     7766, Update Released:                  1088
  Active Connections:          3, Completed Connections:            7183
  Drops:                       0
  Rejected Connection Counts Due To: (Total: 0)
     Not Registered      :          0,  Keepalive Timeout   :          0
     No License          :          0,  Load Level          :          0
     Connection Limit    :          0,  Rate Limit          :          0       <-----Connection li
     Minimum TFO         :          0,  Resource Manager    :          0
     Global Config       :          0,  TFO Overload        :          0
     Server-Side         :          0,  DM Deny             :          0
     No DM Accept        :          0
. . .
```

The No. of active connections field reports the number of connections that are currently being optimized.

In the Policy Engine Statistics section of the output, the Rejected Connection Counts section show various reasons why connections have been rejected. The Connection Limit counter reports the number of times that a connection has been rejected because the maximum number of optimized connections has been exceeded. If you see a high number here, you should look into overload conditions. See the article Troubleshooting

Overload Conditions for more information.

Additionally, TFO optimization for connections that are pushed down from other AOs because they cannot optimize the traffic is handled by the generic AO, which is covered in the article Troubleshooting the Generic AO.

You can view TFO connection statistics by using the **show statistics connection** command. For details on using this command, see the section "Checking the Optimized TCP Connections" in the Troubleshooting Overload Conditions article.

DRE Troubleshooting

When application acceleration is expected but not being observed, verify that the appropriate optimizations are being applied to the traffic flow and that the DRE cache is reducing the size of the optimized traffic appropriately.

Policy engine maps for DRE and LZ optimization include the following:

- DRE + LZ (full): policy-engine application map other optimize full
- DRE only: policy-engine application map other optimize DRE yes compression none
- LZ only: policy-engine application map other optimize DRE no compression LZ
- TFO pass-through: policy-engine application map other pass-through

Various conditions could cause DRE and/or LZ not to be applied to a connection, even though it is configured:

- Cache initialization is in progress
- Disk I/O errors
- Low memory
- Data is not compressible or gain is too small
- Data is encrypted such that it does not contain repeated byte sequences
- Messages are too small to benefit from compression

Note: In all of the above conditions, the **show statistics connection** command will report Acceleration of "TDL" for connections where this was the negotiated policy. Looking at the amount of DRE or LZ bypass traffic will tell you whether DRE or LZ optimizations were actually applied. Use the **show statistics connection conn-id** command, as described later, and look at the DRE encode numbers to see if the DRE or LZ ratio is near 0% and most of the traffic is bypassed. The first three conditions will be reported by the "Encode bypass due to" field and the last three conditions result from the traffic data pattern and are accounted for in the reported DRE and LZ ratios.

You can view the statistics for a specific connection to determine what basic optimizations were configured, negotiated with the peer, and applied by using the **show statistics connection conn-id** command. First you will need to determine the connection ID for a particular connection by using the **show statistics connection** command, as follows:

```
WAE#show stat conn

Current Active Optimized Flows:                       1
  Current Active Optimized TCP Plus Flows:            0
  Current Active Optimized TCP Only Flows:            1
  Current Active Optimized TCP Preposition Flows:     0
Current Active Auto-Discovery Flows:                  0
Current Reserved Flows:                               10
```

```
Current Active Pass-Through Flows:                  0
Historical Flows:                                   375

D:DRE,L:LZ,T:TCP Optimization RR:Total Reduction Ratio
A:AOIM,C:CIFS,E:EPM,G:GENERIC,H:HTTP,M:MAPI,N:NFS,S:SSL,V:VIDEO

ConnID         Source IP:Port          Dest IP:Port            PeerID Accel RR
   343      10.10.10.10:3300    10.10.100.100:80   00:14:5e:84:24:5f T     00.0%        <------
```

You will find the connection IDs for each connection listed at the end of the output. To view the statistics for a specific connection, use the **show statistics connection conn-id** command, as follows:

```
WAE# sh stat connection conn-id 343

Connection Id:            343
   Peer Id:                  00:14:5e:84:24:5f
   Connection Type:          EXTERNAL CLIENT
   Start Time:               Tue Jul 14 16:00:30 2009
   Source IP Address:        10.10.10.10
   Source Port Number:       3300
   Destination IP Address:   10.10.100.100
   Destination Port Number:  80
   Application Name:         Web                                          <-----Application n
   Classifier Name:          HTTP                                         <-----Classifier na
   Map Name:                 basic
   Directed Mode:            FALSE
   Preposition Flow:         FALSE
   Policy Details:
          Configured:        TCP_OPTIMIZE + DRE + LZ                      <-----Configured po
             Derived:        TCP_OPTIMIZE + DRE + LZ
                Peer:        TCP_OPTIMIZE + DRE + LZ
          Negotiated:        TCP_OPTIMIZE + DRE + LZ                      <-----Policy negoti
             Applied:        TCP_OPTIMIZE + DRE + LZ                      <-----Applied polic
. . .
```

The Application Name and Classifier Name fields tell you the application and classifier applied to this connection.

The optimization policies are listed in the Policy Details section. If the Configured and Applied policies do not match, it means that you configured one policy for this type of connection but a different policy was applied. This could result from the peer being down, misconfigured, or overloaded. Check the peer WAE and its configuration.

The following section of output shows DRE encode/decode-related statistics including the number of messages, how many had DRE applied, LZ applied, or bypassed DRE and LZ:

```
. . .
DRE: 353

Conn-ID: 353 10.10.10.10:3304 -- 10.10.100.100:139  Peer No:  0 Status: Active
--------------------------------------------------------------------------------
Open at 07/14/2009 16:04:30, Still active
Encode:
   Overall: msg:        178, in:  36520 B, out:   8142 B, ratio:  77.71%         <-----Overall co
       DRE: msg:          1, in:    356 B, out:    379 B, ratio:   0.00%         <-----DRE compre
DRE Bypass: msg:        178, in:  36164 B                                        <-----DRE bypass
        LZ: msg:        178, in:  37869 B, out:   8142 B, ratio:  78.50%         <-----LZ compres
 LZ Bypass: msg:          0, in:      0 B                                        <-----LZ bypass
    Avg latency:      0.335 ms     Delayed msg:          0                     <-----Avg latenc
  Encode th-put:    598 KB/s                                                    <-----In 4.3.3 a
  Message size distribution:
```

```
    0-1K=0%  1K-5K=0%  5K-15K=0%  15K-25K=0%  25K-40K=0%  >40K=0%                <-----In 4.3.3 a
Decode:
   Overall: msg:      14448, in:   5511 KB, out:    420 MB, ratio:  98.72%      <-----Overall co
      DRE: msg:      14372, in:   5344 KB, out:    419 MB, ratio:  98.76%      <-----DRE compre
DRE Bypass: msg:      14548, in:    882 KB                                      <-----DRE bypass
       LZ: msg:      14369, in:   4891 KB, out:   5691 KB, ratio:  14.07%      <-----LZ compres
 LZ Bypass: msg:         79, in:    620 KB                                      <-----LZ bypass
    Avg latency:      4.291 ms                                                 <-----Avg latenc
  Decode th-put:   6946 KB/s                                                   <-----In 4.3.3 a
  Message size distribution:
    0-1K=4%  1K-5K=12%  5K-15K=18%  15K-25K=9%  25K-40K=13%  >40K=40%         <-----Output from he
. . .
```

The following statistics are highlighted in the above example for both encoding and decoding:

- Overall ratio - the overall compression ratio for the data including both DRE and LZ
- DRE ratio - the compression ratio due to DRE alone
- DRE Bypass - the number of messages and bytes that bypassed DRE
- LZ ratio - the compression ratio due to LZ alone
- LZ Bypass - the number of messages and bytes that bypassed LZ
- Avg latency - the average latency for the encode or decode operation

If you see a large amount of bypass traffic, the DRE compression ratio will be smaller than expected. It could be due to encrypted traffic, small messages, or otherwise uncompressible data. Consider contacting TAC for further troubleshooting help.

If you see a large amount of LZ Bypass traffic, this could be due to a large amount of encrypted traffic, which is not generally compressible.

The Average latency numbers can be useful for debugging a throughput issue. Depending on the platform, both the encode and decode average latency are typically in the single digits of ms. If users experience low throughput and one or both of these numbers are higher, it indicates an issue with encoding or decoding, generally on the side with the higher latency.

It may be useful to look at the DRE statistics data such as the oldest usable data, cache size, percent of cache used, hash table RAM used, and so on by using the **show statistics dre detail** command, as follows:

```
WAE# sh stat dre detail

Cache:
    Status: Usable, Oldest Data (age): 10h                    <-----Cache age
    Total usable disk size: 311295 MB,  Used: 0.32%           <-----Percent cache used
      Hash table RAM  size:   1204 MB,  Used: 0.00%           <-----Output from here is in 4.3
. . .
```

If you are not seeing significant DRE compression, it could be because the DRE cache is not populated with enough data. Check if the cache age is short and less than 100 percent of the cache is used, which would indicate this situation. The compression ratio should improve as the cache fills with more data. If 100% of the cache is used and the cache age is short, it indicates that the WAE may be undersized and not able to handle the traffic volume.

If you are not seeing significant DRE compression, look at the Nack/R-tx counters in the following section of command output:

```
Connection details:
 Chunks: encoded 398832,  decoded 269475,  anchor(forced) 43917(9407)     <-----In 4.3.3 and ear
 Total number of processed messges: 28229                                 <-----In 4.3.3 and ear
```

```
 num_used_block per msg: 0.053597                                          <-----In 4.3.3 and ear
Ack: msg 18088,  size  92509 B                                             <-----In 4.3.3 and ear
Encode bypass due to:                                                      <-----Encode bypass re
    remote cache initialization: messages: 1,  size:    120 B
    last partial chunk: chunks: 482,  size:  97011 B
    skipped frame header: messages: 5692,  size:    703 KB
Nacks: total 0                                                             <-----Nacks
R-tx: total 0                                                              <-----Retransmits
Encode LZ latency:      0.133 ms per msg
Decode LZ latency:      0.096 ms per msg
. . .
```

The Nacks and R-tx counters should generally be low relative to the traffic volume. For example, about 1 per 100 MB of original (unoptimized) traffic. If you see significantly higher counts, it could indicate a DRE cache synchronization problem. Use the **clear cache dre** command to clear the DRE cache on all devices, or contact TAC.

The encode bypass reasons counters report the number of bytes bypassed due to various reasons. This can help you determine what is causing bypass traffic (other than an unoptimizable data pattern).

It is sometimes helpful to identify the connected and active peer WAEs and look at peer statistics, which you can do with the **show statistics peer dre** command as follows:

```
WAE# sh stat peer dre

Current number of connected peers:  1
Current number of active peers:     1
Current number of degrade peers:    0
Maximum number of connected peers:  1
Maximum number of active peers:     1
Maximum number of degraded peers:   0

Active peer details:

Peer-No : 0                    Context: 65027
Peer-ID : 00:14:5e:95:4a:b5
Hostname: wae7.example.com                                      <-----Peer hostname
--------------------------------------------------------------------------------

Cache:  Used disk: 544 MB, Age: 14d23h                          <-----Peer cache details in
Cache:  Used disk: 544 MB                                       <-----Peer cache details in
Peer version:  0.4                                              <-----
Ack-queue size:   38867 KB                                           |
Buffer surge control:                                                |<---In 4.3.3 and earli
  Delay:   avg-size          0 B, conn:          0, flush:          0  |
  Agg-ft:  avg-size      20902 B, conn:        388, flush:          0  |
           remote low-buff:  0, received flush:          0            <-----

Connections:   Total (cumulative): 3226861,  Active: 597
Concurrent Connections (Last 2 min): max 593, avg 575
. . .
```

Other output from this command shows the encode and decode statistics similar to an individual connection.