

Gateway Proxy debugging

Last updated by | Vitor Tomaz | Feb 18, 2021 at 3:28 AM PST

Contents

- [Gateway proxy debugging](#)
 - [Proxy telemetry](#)
 - [Proxy errors](#)
 - [Proxy failures by target ring](#)
 - [Total TCP Port exhaustion errors/min](#)
 - [TDS 32K Session Limit](#)
 - [Identify proxy disconnects based on Client / Tenant Ring di...](#)

Gateway proxy debugging

Proxy telemetry

When the proxy closes connection, a event is fired. The following are *some* important columns. Not all are mentioned. Please do a `| limit 100` against the table and hide empty columns in kusto to find out the schema, or lookup the xevent definition. Server_name: The FQDN of server_name as found in the login packet. It contains the protocol name and port number (if any was entered in connection string) along with DNS Name.

Columns:

- tcp:seahawks.database.windows.net,1433
- logical_server_name: The server's short name. Example: seahawks
- Database_name
- Connection_id (Same as the connection_id in corresponding process_login_finish)
- Bytes_Read - total bytes read (from backend server by gw) during the duration of the connection.
- Bytes_written - total bytes written (to backend server by gw) during the duration of the connection.
- Packets_Read - total packets read from backend server by gw during the duration of the connection.
- Packets_written - total packets written to backend server by gw during the duration of the connection.
- Buffer_size - connection buffer size that the clients have set in the connection string (PacketSize is the name in [ADO.NET](#) [🔗](#))
- Duration_in_ms - the total duration in milliseconds that the proxy connection session was active for.
- Is_azure_ip - true (or 1), if the peer_address is an azure ip. 0 if it is not.
- Proxy_override behavior - determines what the proxy override behavior is as set in CMS.

proxy_override value of e_crContinue would imply eProxyOverrideBehaviorDefault (default behavior)

```
MonLogin | where event == "proxy_close_connection" and originalEventTimestamp > ago(6h) and ClusterName contain
| limit 1000
```

Proxy errors

Process_login_finish state 22 means Proxy failure. To discover error code look for login_substep_finish event under the same connection_id

- 10060 - TCP Timeout (suspect xdbhost)
- 10055 - TCP Memory error, most likely TCP port exhaustion (due to Windows/Helper)
- 1460 - Login spikes (throttle) causing failures (Login was disconnected because it was in the queue for too long)

```
MonLogin
| where state == 22 and package == "xdbgateway"
| where event == "process_login_finish"
| where originalEventTimestamp > ago(1d)
```

Proxy failures by target ring

```
MonLogin
| where originalEventTimestamp >= ago(1d)
| extend TargetRing = toupper(extract("[a-f0-9]*.(tr\\d+).(*)",1,instance_name) )
| where package == "xdbgateway"
| where error == 40613 and state==22
| where event contains "process_login_finish"
| summarize count() by TargetRing, bin(originalEventTimestamp, 1m)
| render timechart
```

% of proxy failures/min due to TCP Port exhaustion (w.r.t. to total proxy attempts, no total logins)

```
MonLogin
| where (event == "login_substep_failure" or event == "process_login_finish") and NodeRole == "GW"
| extend is_good = iff(event == "process_login_finish" and result == "e_crContinueSameState", 1, 0)
| extend is_bad = iff(error_code == 10055, 1, 0)
| summarize sum(is_bad), sum(is_good) by originalEventTimestamp bin=time(1m)
| extend pct = (sum_is_bad / (sum_is_bad + sum_is_good) * 1.0) * 100
| project originalEventTimestamp, pct
```

Total TCP Port exhaustion errors/min

```
MonLogin
| where event == "login_substep_failure" and NodeRole == "GW"
| where error_code == 10055
| summarize cnt() by originalEventTimestamp bin=time(1m)
```

TDS 32K Session Limit

```
MonLogin  
| where sni_consumer_error==17809  
| where event=="connectivity_ring_buffer_recorded"
```

Identify proxy disconnects based on Client / Tenant Ring disconnects.

```
let svr = "psqlaaep001";  
let db = "EDWDataMartAir";  
MonLogin  
//| where originalEventTimestamp between ((tm-2h)..(tm+1h))  
| where originalEventTimestamp >= ago(7d)  
| where logical_server_name =~ svr and database_name =~ db  
| where event == "proxy_close_connection"  
| where package == "xdbgateway"  
| extend disconnect_from = iff(extra_info contains "Svr", "Tr", "client")  
| summarize count() by disconnect_from, bin(originalEventTimestamp, 1h)  
| render timechart
```

How good have you found this content?



-