

---

Article

[Tani Frankel](#) · Sep 7, 2020 2m read

## How to Prevent Superfluous Connection Logging in TCP Interfaces

You might have encountered this and not known how to prevent it, or perhaps not even noticed it...

But when you have TCP Interoperability Production business components (e.g. an HL7 business service) you probably have (depending on your [StayConnected](#) setting) many entries in the Event Log of Info type. Each entry would log a connection or disconnection.

For example:

Type	ID	Time Logged	Session	Job	Source	Text
Info	823	2020-09-07 15:15:06.929		19020	From TCP	New connection on port 59999 from 127.0.0.1:52002
Info	820	2020-09-07 15:15:01.905		19020	From TCP	New connection on port 59999 from 127.0.0.1:52001
Info	817	2020-09-07 15:14:56.891		19020	From TCP	New connection on port 59999 from 127.0.0.1:52000

And:

Type	ID	Time Logged	Session	Job	Source	Text
Info	824	2020-09-07 15:15:06.943		19020	From TCP	Disconnecting from 59999<-127.0.0.1:52002
Info	821	2020-09-07 15:15:01.918		19020	From TCP	Disconnecting from 59999<-127.0.0.1:52001
Info	818	2020-09-07 15:14:56.908		19020	From TCP	Disconnecting from 59999<-127.0.0.1:52000

This information can indeed be handy for debugging for example or keeping an eye on how the connections are working, but on an ongoing basis, especially if StayConnected is not permanent (-1) then this could generate a very large amount of entries.

If you would like to avoid this kind of logging (and actually turn the entries into configurable Trace events, instead of the Info ones), you can do this by performing the steps described below.

Some background first -

The code in the TCP Adapter, responsible for this logging, is conditional (to log an INFO kind of event or a TRACE one) - depending on a property in the Adapter called [%logconnections](#).

The default value for this property is 1 (one), i.e. to log these connect and disconnect event (as INFO events).

At the startup of the Inbound or Outbound TCP Adapter (in the OnInit() method) the Adapter checks for a value of a Class Parameter in the host's class (e.g. the Business Service or Business Operation class). The name of this Class Parameter is LOGCONNECTIONS.

For example:

```
If 0=..BusinessHost.%GetParameter("LOGCONNECTIONS") Set ..%logConnections=0
```

So in order to prevent the events to be logged (as INFO events) you can simply create a Class Parameter called LOGCONNECTIONS in your business host class and set it's value to 0 (zero).

Assuming for example you have an HL7 inbound Business Service the steps would be -

1. Create a Class that extends EnsLib.HL7.Service.TCPService
2. Add a Class Parameter called LOGCONNECTIONS with a value of 0
3. Change your Business Service's class in your Production to be your newly created class.

Here's a sample class per above -

```
Class Test.BS.HL7.TCPService Extends EnsLib.HL7.Service.TCPService
{
    Parameter LOGCONNECTIONS As %Boolean = 0;
}
```

And in my Production's Business Service I have:

The screenshot shows the 'Production Configuration' window for a 'From TCP' service. The 'Settings' tab is selected. Under 'Informational Settings', the 'Class Name' field is highlighted with a blue box and contains the text 'Test.BS.HL7.TCPService'. Other fields include 'Comment', 'Category', 'Description', 'Adapter Class Name' (set to 'EnsLib.HL7.Adapter.TCPInboundAdapter'), and 'Adapter Description' (set to 'Inbound adapter for HL7 v2 over TCP').

Now connections will not be logged at all.

Or, if you need to see the logs for debugging or any other purpose, turn on tracing for your component and you will see now Trace events (instead of the Info ones we had before).

For example:

Type	ID	Time Logged	Session	Job	Source	Text
Trace	937	2020-09-07 15:34:44.365		11312	From TCP	New connection on port 59999 from 127.0.0.1:52978
Trace	913	2020-09-07 15:34:39.345		11312	From TCP	New connection on port 59999 from 127.0.0.1:52773
Trace	885	2020-09-07 15:34:34.328		11312	From TCP	New connection on port 59999 from 127.0.0.1:52772

And:

Type	ID	Time Logged	Session	Job	Source	Text
Trace	956	2020-09-07 15:34:44.385		11312	From TCP	Disconnecting from 59999<-127.0.0.1:52978
Trace	932	2020-09-07 15:34:39.366		11312	From TCP	Disconnecting from 59999<-127.0.0.1:52773
Trace	905	2020-09-07 15:34:34.343		11312	From TCP	Disconnecting from 59999<-127.0.0.1:52772

Hopefully this should help you keep your Event Log cleaner and at a better size.

[#Business Operation](#) [#Business Service](#) [#Debugging](#) [#HL7](#) [#Interoperability](#) [#Tips & Tricks](#) [#Ensemble](#)  
[#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)

---

Source

URL:<https://community.intersystems.com/post/how-prevent-superfluous-connection-logging-tcp-interfaces>