

## OpenOpcUa - Anomalie #18

### unable to connect to OpenOpcUa server with Inductive Automation Ignition

02/22/2014 12:07 AM - Peter Speybrouck

<b>Status:</b>	Rejeté	<b>Start date:</b>	02/21/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Michel Condemine Admin	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Description</b>			
<p>While testing a few OPC UA servers, I tried to link OpenOpcUA to Ignition by Inductive Automation.</p> <p>In this case, Ignition opc ua client is trying to connect to the OpenOpcUa server, both running on the same machine (Windows 7, 64 bit). I have the same issue when connecting to a remote OpenOpcUa server on a Raspberry PI though.</p> <p>The connection fails and Ignition is reporting a nullreference error in GetEndpoints: @java.lang.NullPointerException: null com.inductiveautomation.xopc.client.util.ConnectionUtil.getEndpoint(ConnectionUtil.java:78) com.inductiveautomation.xopc.client.OpcUaConnection.createUaClient(OpcUaConnection.java:523) com.inductiveautomation.xopc.client.OpcUaConnection.verifyConnectionState(OpcUaConnection.java:141) com.inductiveautomation.xopc.client.FailoverAwareOpcUaConnection.verifyConnectionState(FailoverAwareOpcUaConnection.java:75) com.inductiveautomation.ignition.gateway.opc.OPCManagerImpl\$ServerWrapper.verifyConnectionState(OPCManagerImpl.java:742) com.inductiveautomation.ignition.gateway.opc.OPCManagerImpl\$OPCConnectionKeepalive.run(OPCManagerImpl.java:618) com.inductiveautomation.ignition.common.execution.impl.BasicExecutionEngine\$TrackedTask.run(BasicExecutionEngine.java:573) java.util.concurrent.Executors\$RunnableAdapter.call(Executors.java:471) java.util.concurrent.FutureTask\$Sync.innerRunAndReset(FutureTask.java:351) java.util.concurrent.FutureTask.runAndReset(FutureTask.java:178) java.util.concurrent.ScheduledThreadPoolExecutor\$ScheduledFutureTask.access\$301(ScheduledThreadPoolExecutor.java:178) java.util.concurrent.ScheduledThreadPoolExecutor\$ScheduledFutureTask.run(ScheduledThreadPoolExecutor.java:293) java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1110) java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:603) java.lang.Thread.run(Thread.java:722)@</p> <p>I have increased logging in both servers (attached) but it is not clear to me yet whether the problem is with Ignition, OpenOpcUa or the configuration to connect both.</p> <p>For kepware, Inductive automation recommends using the IP address instead of localhost or the hostname but I tried several configurations.</p> <p>A comment by an Ignition forum moderator: @GetEndpointsResponse contains a null EndpointDescription list and Ignition isn't expecting that to be null... @</p> <p>From debugging OpenOpcUa, it seems to me that it is sending an Endpoint description. Maybe it is not in a format that Ignition expects, but since for example UAExpert opc UA client has no trouble I am wondering if it is not Ignition that is expecting a value somewhere where there is not necessarily something needed...</p>			

#### History

##### #1 - 02/22/2014 12:24 AM - Michel Condemine Admin

- Assignee set to Michel Condemine Admin

Peter,  
the Ignition client is trying to connect to opc.tcp://192.168.0.101:16664 this is not the URL you setup.  
You configure OpenOpcua to listen on opc.tcp://peter:16664/embedded  
So let change to ignition configuration to connect to opc.tcp://192.168.0.101:16664/embedded.

Hope it helps  
Michel

##### #2 - 02/22/2014 02:03 AM - Peter Speybrouck

Michel,

Actually OpenOpcUa is constructing that URL from my hosts file. It had some 3dns.adobe.com entry which was causing the weird opc.tcp://3dns.adobe.com:16664/4CEUAServer URL. After cleaning up the hosts file, it took my laptops hostname (peter).

I tried configuring Ignition to connect to opc.tcp://peter:16664. I can't point it directly to opc.tcp://peter:16664/embedded , I can only give a hostname and a port. (I think)

I also tried to force OpenOpcUa to construct the url opc.tcp://192.168.0.101:16664/embedded using the hosts file with as first line "127.0.0.1 192.168.0.101" but I am still getting the same errors.

The Ignition client certificate is also in the CertificateStore\certs directory.

### **#3 - 02/22/2014 02:14 AM - Michel Condemine Admin**

Peter,  
In the OpcOpcUaCoreServer the endpoint url is build from the information available in the XML configuration file.  
Let say you have :

the endpoint url will be opc.tcp://localhost:16664/4CEUAServer.  
The client MUST use this EndPoint url to connect. So you just have to configure Ignition to force him to use the correct Endpoint url.

Regards  
Michel

### **#4 - 02/22/2014 12:30 PM - Peter Speybrouck**

Michel,  
Thank you for the clarification. I will try and force this. Ignition does have an option to override the host name:  
@If specified, the host in the endpoint URL returned by the server will be ignored and the override will be used in its place.@

However, while debugging, I noticed that not all information is coming from the XML config file.  
In the example you give here, the "localhost" is fetched from the hosts file from the operating system.  
To block certain domain names, I had some non standard entries in there and the first one was used to construct the endpoint URL. In my case that was not localhost.

The part of the code where it gets that entry is the "getnameinfo" call in OpenOpcUaSharedLib/source/opcu\_certificates.cpp, line 2882.

If I modify the hosts file, the endpoint url changes (if I remove the generated certificate).

Anyway, I will experiment some more and keep you updated.

### **#5 - 02/22/2014 01:40 PM - Michel Condemine Admin**

Peter,  
To clarify, let me suggest the following :  
1- Configure the IP address instead of the Hostname. Both works but it will remove resolution problems  
2- You have to use full Endpoint url ie:opc.tcp://192.168.0.1:16664/4CEUAServer

Remember that is project is based on sponsoring and feedback.

Regards  
Michel Condemine  
OpenOpcUa Project Leader  
+33 (0)4 67 79 07 37

### **#6 - 03/13/2014 01:07 AM - Peter Speybrouck**

- File wiresharkLog.pcapng added

Michel,  
just to add some more information:  
I understand ignition or any other client should use the correct endpoint url to set up a connection, but it is actually failing one step before that I think.

Using wireshark (which has a opc ua decoder) I see the getEndpointsRequest and the getEndpointsResponse.

The response contains an empty array of endpoints so ignition does not know which endpoint url to connect to.

If my understanding is correct, ignition is requesting the endpoint urls with this request (and is asking them on opc.tcp://:16664 since it does not have the endpoint url yet) and is expecting to get the full endpoint URL you mentioned (like opc.tcp://192.168.0.1:16664/4CEUAServer) to set up a secure channel and session.

I'll see if I can debug this a little or even get it working.

attached is an example log captured with wireshark.

**#7 - 03/13/2014 02:09 AM - Michel Condemine Admin**

Peter,  
The Ignition Moderator is right. The server is not suppose to provide a null EndPoint Description.  
But the client is suppose to use the correct endpoint in its request. The Ignition client *is not using* the EndPoint URL of the server.  
Ignition client request for *opc.tcp://moefie:16664* so the answer is empty because based on your configuration the correct endpoint url is *opc.tcp://moefie:16664/embedded*

I hope it helps

Regards  
Michel

**#8 - 02/24/2015 02:19 PM - Michel Condemine Admin**

- *Status changed from Nouveau to Rejeté*

Ignition use -1 instead of 0 the tells that a array is empty. (ProfileUris on GetEndpoints)  
This was confusing for the server. I put a workaround and everything works fine now.  
Ignition should fix his client.  
use OpenOpcUa >=1.0.2.3 to access this fix.

**Files**

debuglogs.zip	1.28 MB	02/21/2014	Peter Speybrouck
wiresharkLog.pcapng	10.7 KB	03/13/2014	Peter Speybrouck