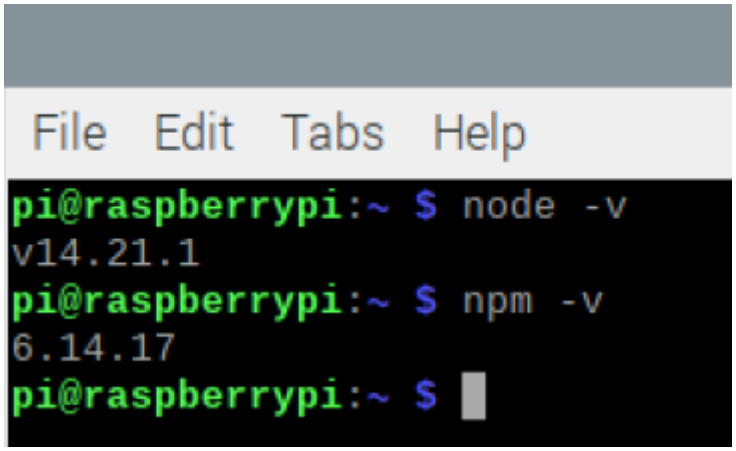
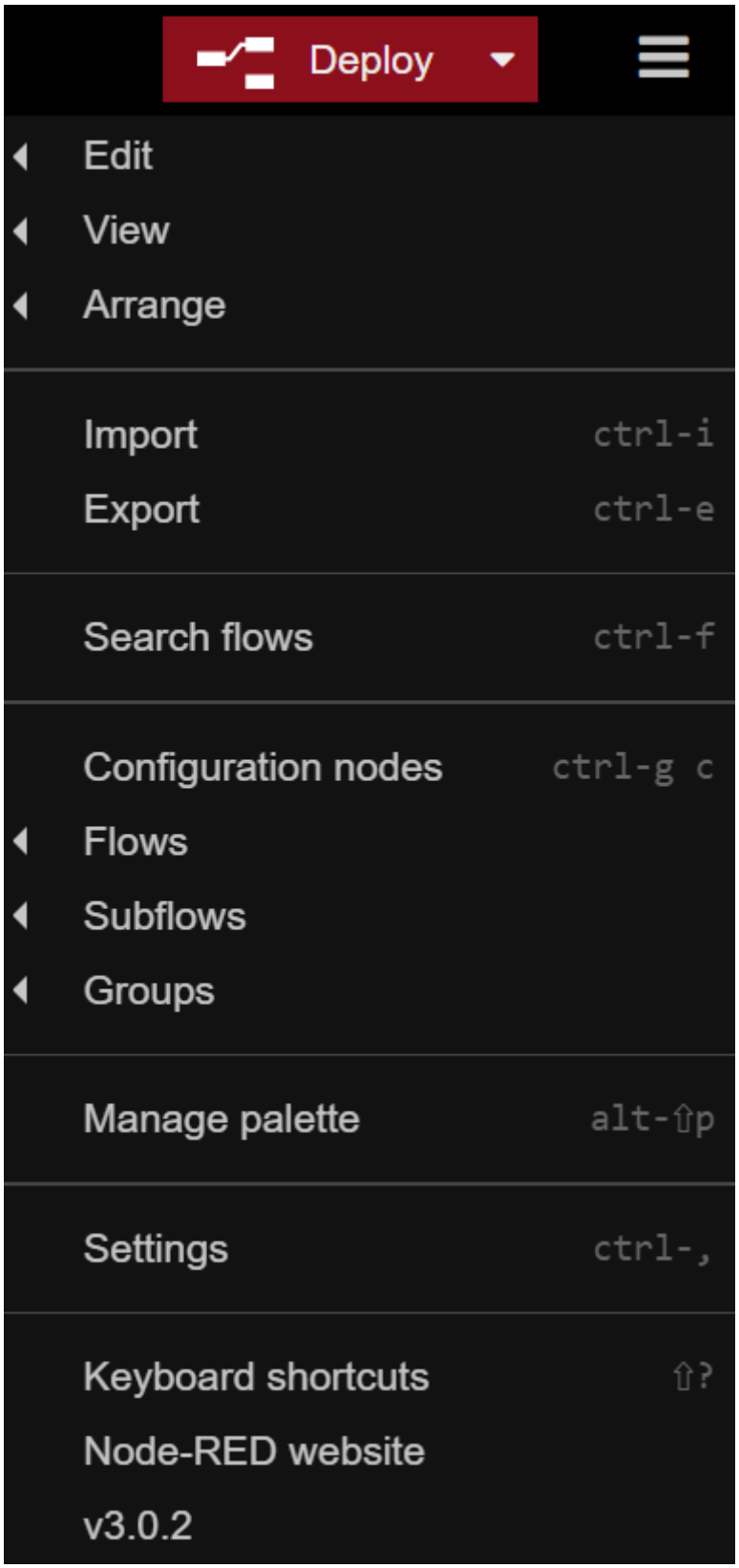
First steps with Node-RED OPC-UA server, and EtherNet/IP

The following versions have been used



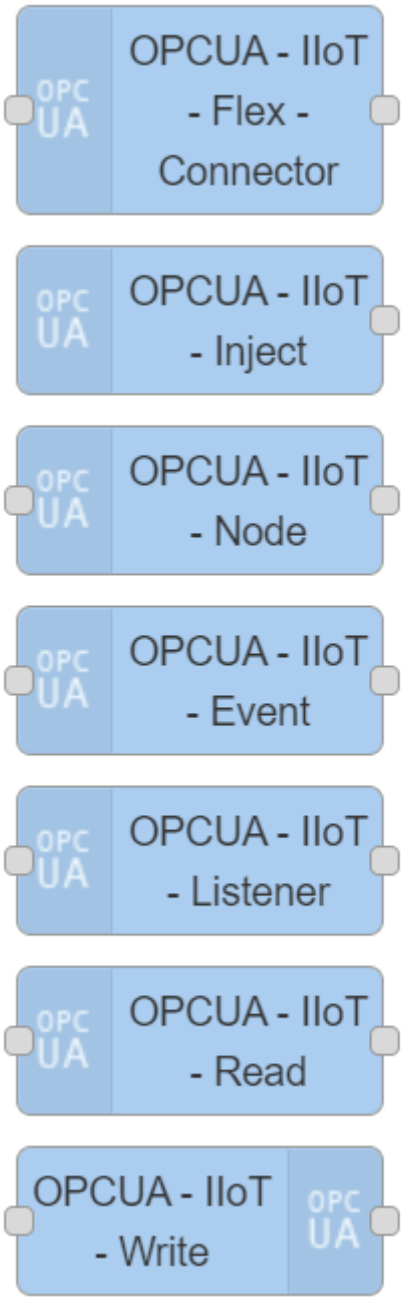
Node-RED v3.0.2



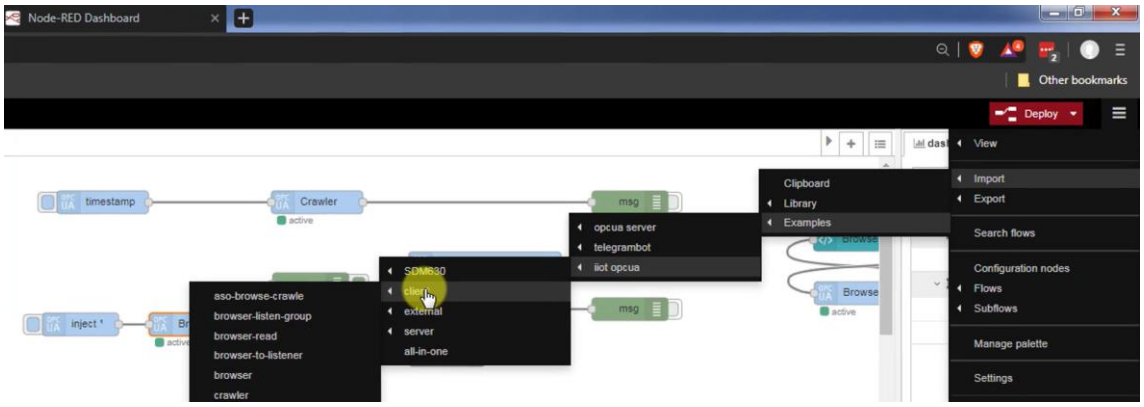
Let’s install these nodes:

Nodered-contrib-iiot-opcua

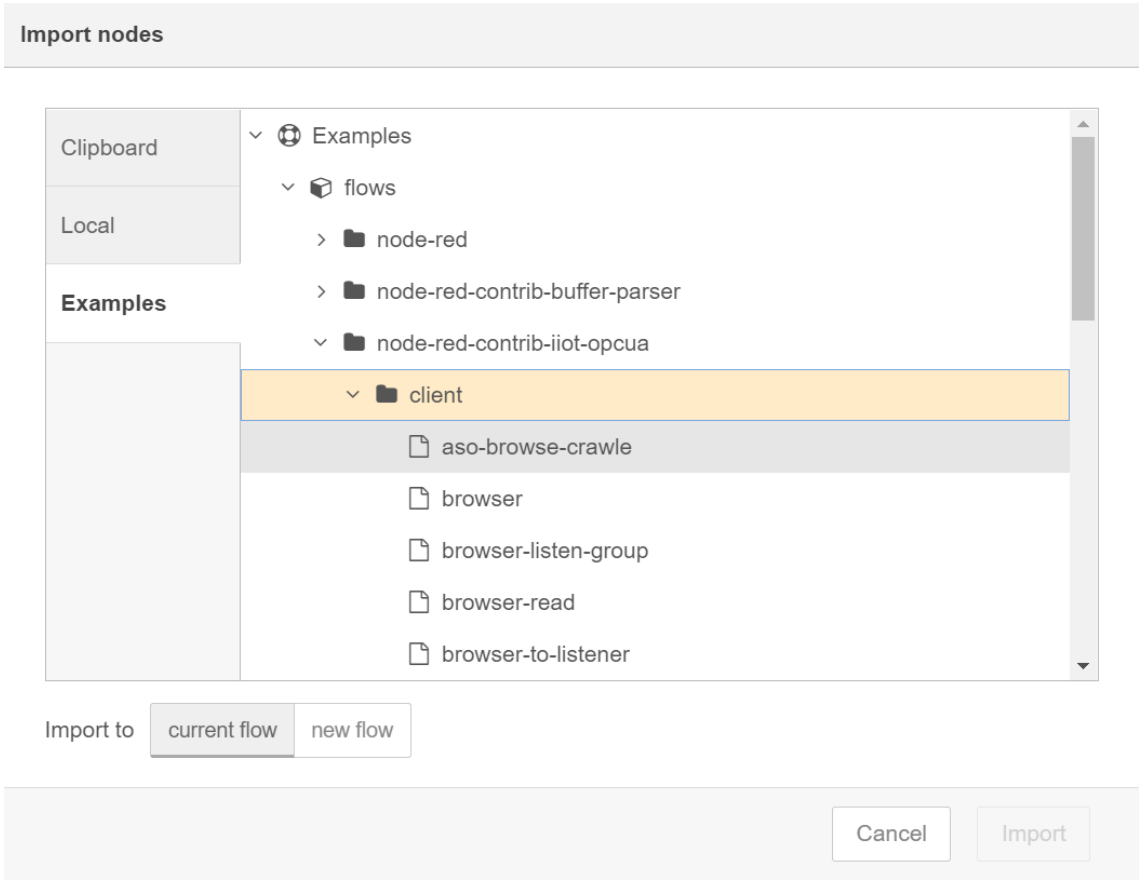
node-red-contrib-cip-ethernet-ip

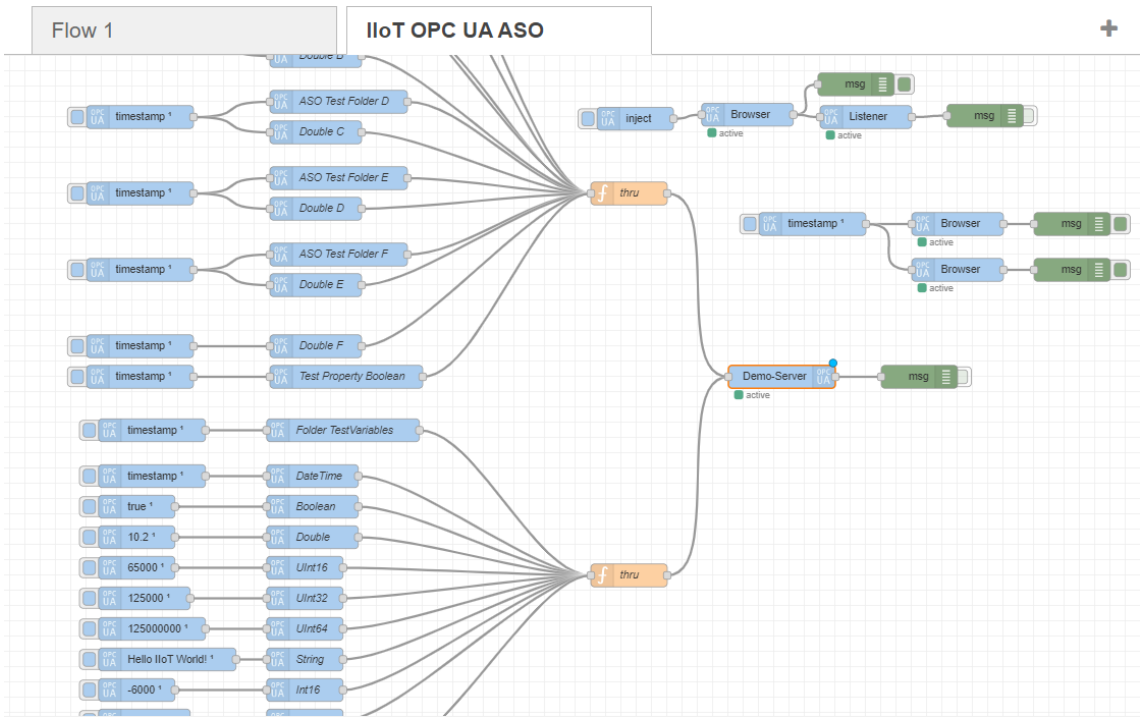


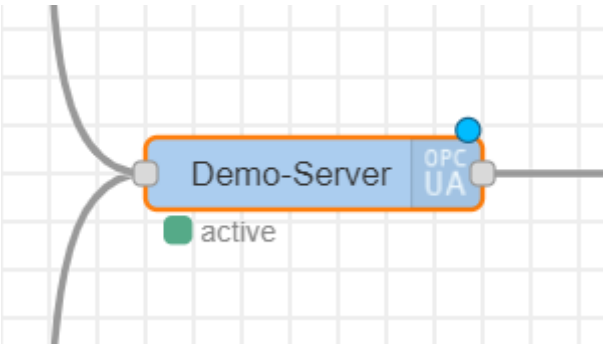
Next import following example on the working area: aso-browse-crawle



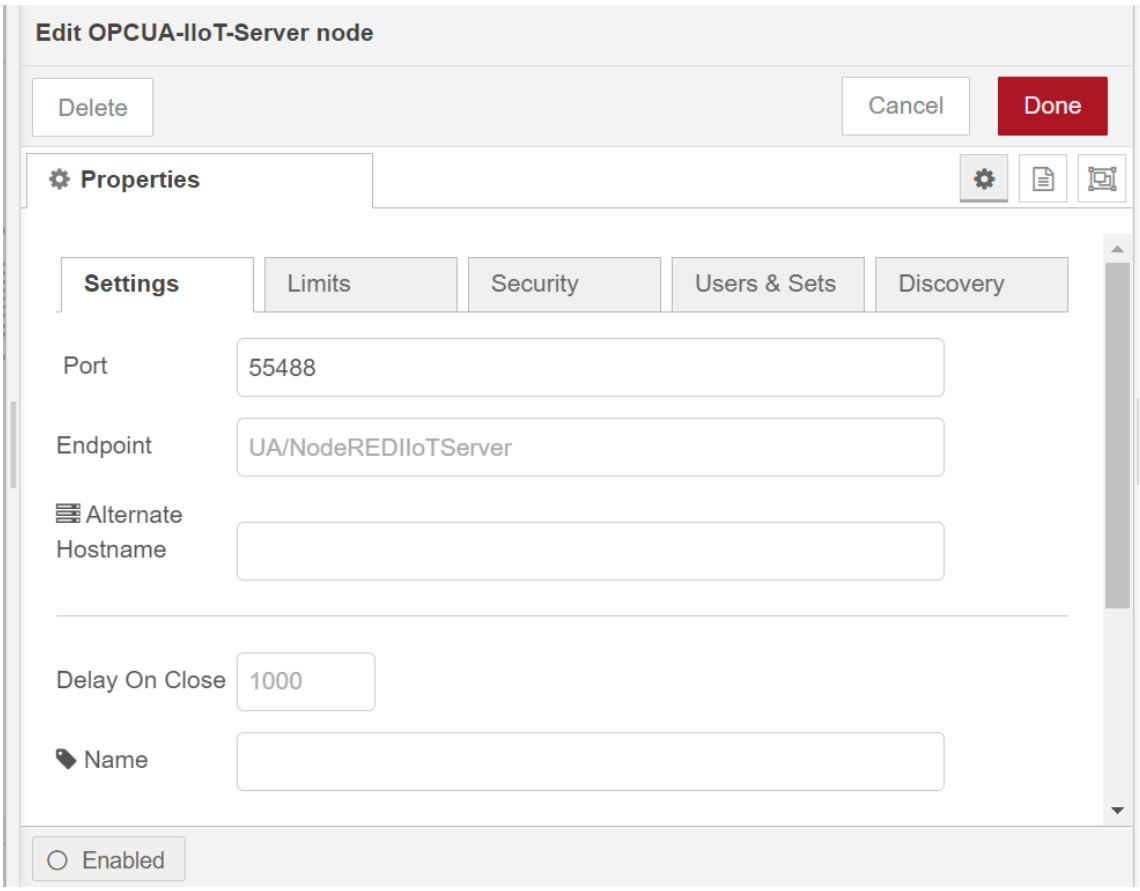
Or depending on the Node-red version you have you will find this way


Import nodes
Clipboard
Local
Examples
Examples
flows
node-red
node-red-contrib-buffer-parser
node-red-contrib-iiot-opcua
client
aso-browse-crawle
browser
browser-listen-group
browser-read
browser-to-listener
Import to
current flow
new flow
Cancel
Import


Flow 1
IIoT OPC UA ASO
timestamp '
ASO Test Folder D
Double C
ASO Test Folder E
Double D
ASO Test Folder F
Double E
Double F
Test Property Boolean
Folder TestVariables
DateTime
Boolean
Double
UInt16
UInt32
UInt64
String
Int16
true '
10.2 '
65000 '
125000 '
125000000 '
Hello IIoT World! '
-6000 '
thru
inject
Browser
Listener
msg
active
Demo-Server

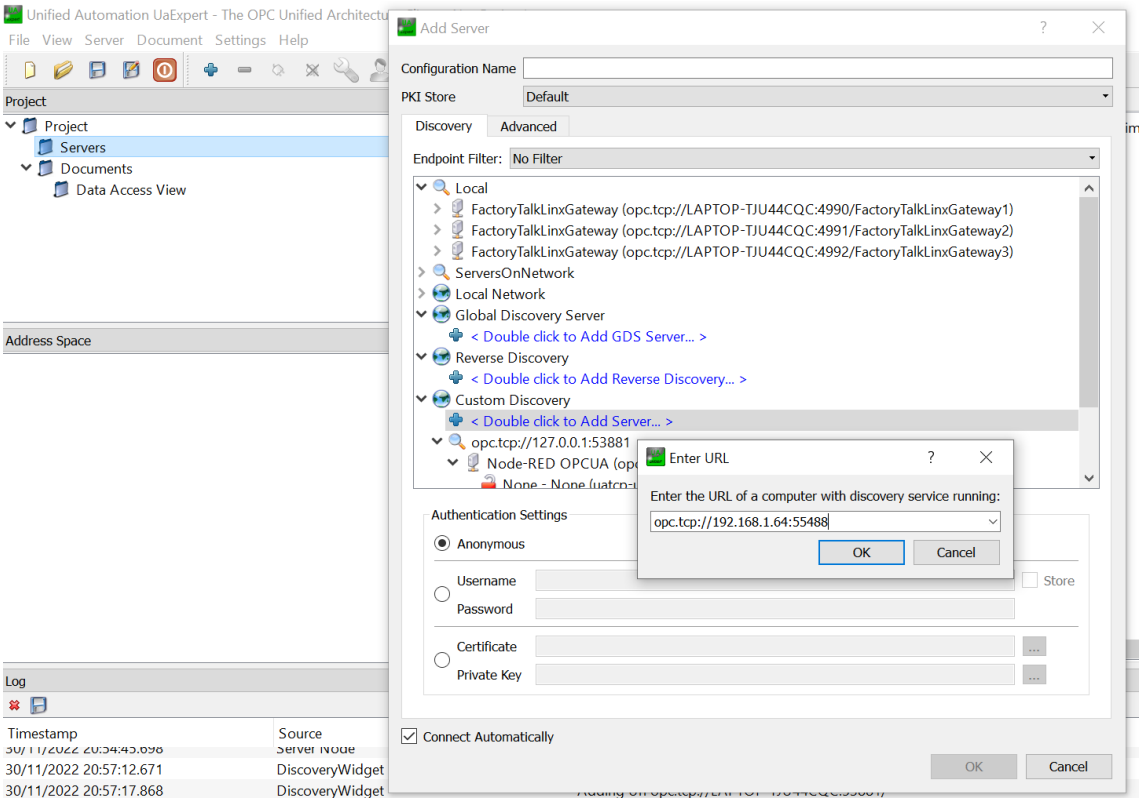


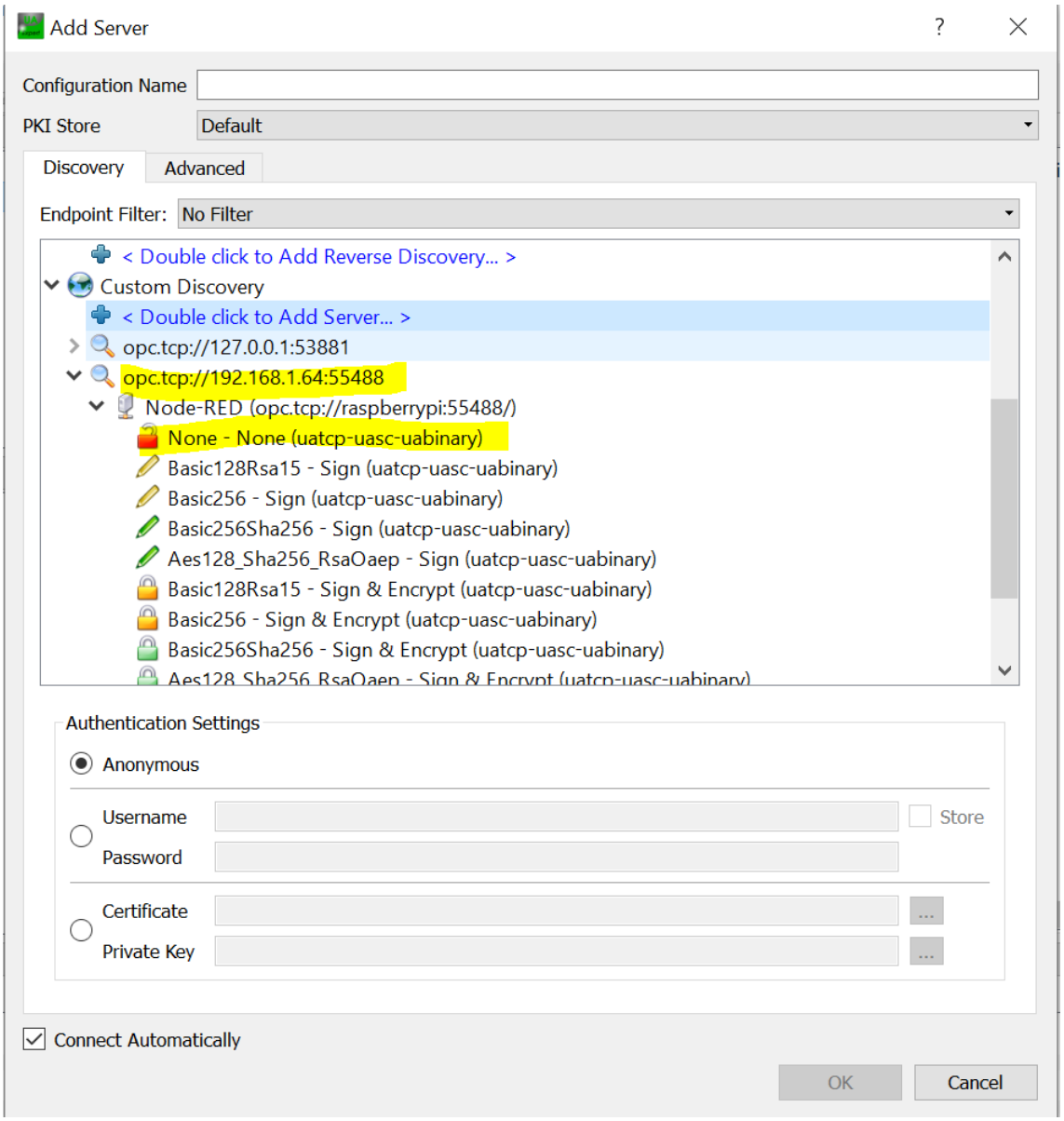
The server is active and listening on port 55488

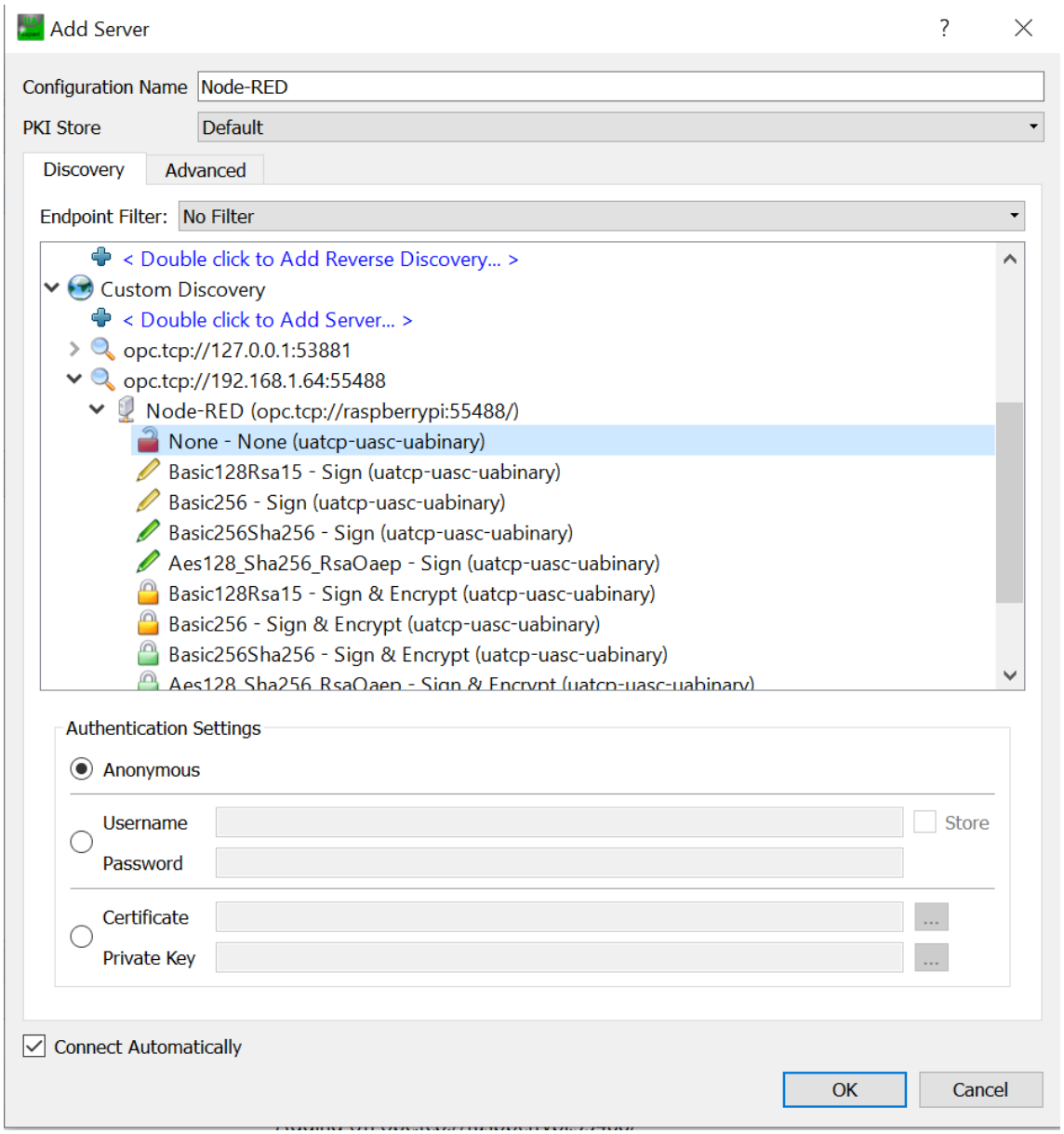


The OPC server is running on IP address 192.168.1.64. This is the wireless IP address got thru DCHP from router.

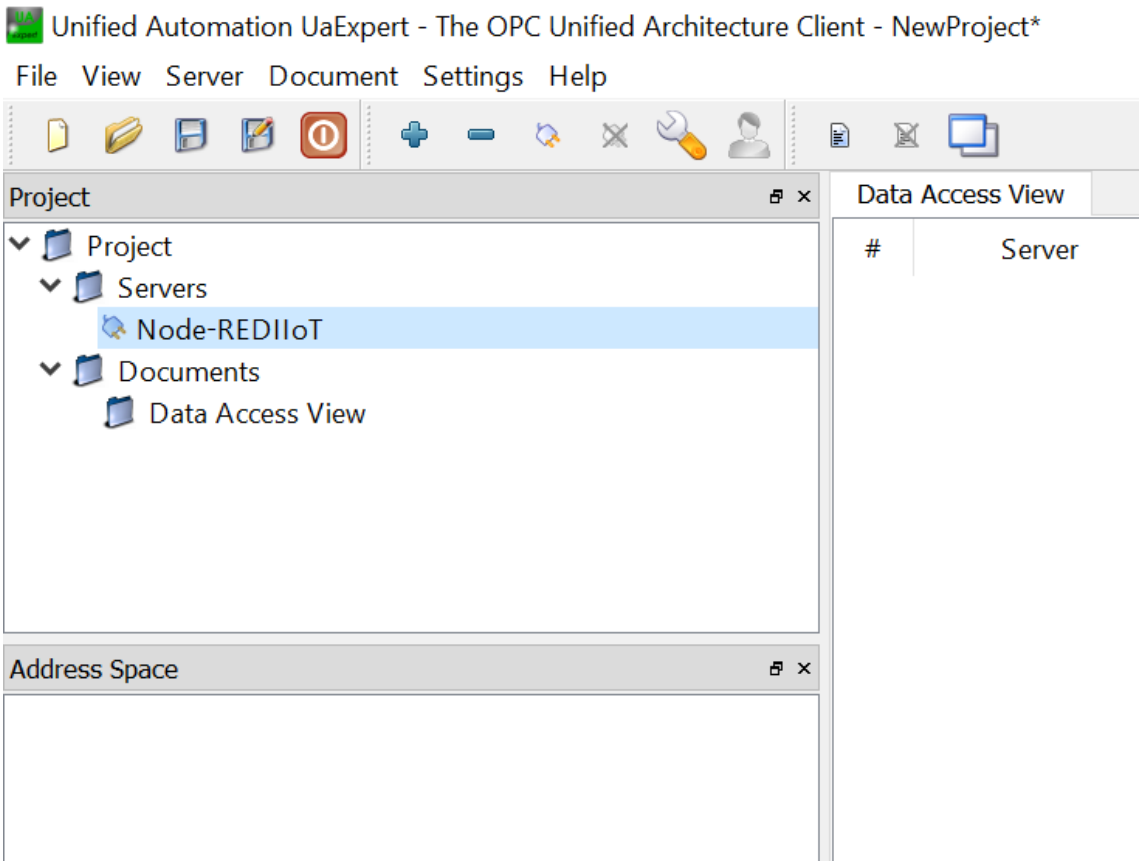
Let's open an OPC UA client. And add a new server.


Unified Automation UaExpert - The OPC Unified Architectu
File View Server Document Settings Help
Project
Project
Servers
Documents
Data Access View
Address Space
Log
Timestamp
Source
30/11/2022 20:57:12.671
DiscoveryWidget
30/11/2022 20:57:17.868
DiscoveryWidget
Add Server
Configuration Name
PKI Store
Default
Discovery
Advanced
Endpoint Filter:
No Filter
Local
FactoryTalkLinxGateway (opc.tcp://LAPTOP-TJU44CQC:4990/FactoryTalkLinxGateway1)
FactoryTalkLinxGateway (opc.tcp://LAPTOP-TJU44CQC:4991/FactoryTalkLinxGateway2)
FactoryTalkLinxGateway (opc.tcp://LAPTOP-TJU44CQC:4992/FactoryTalkLinxGateway3)
ServersOnNetwork
Local Network
Global Discovery Server
< Double click to Add GDS Server... >
Reverse Discovery
< Double click to Add Reverse Discovery... >
Custom Discovery
< Double click to Add Server... >
opc.tcp://127.0.0.1:53881
Node-RED OPCUA (op
Enter URL
Enter the URL of a computer with discovery service running:
opc.tcp://192.168.1.64:55488
OK
Cancel
Authentication Settings
Anonymous
Username
Password
Store
Certificate
Private Key
Connect Automatically
OK
Cancel

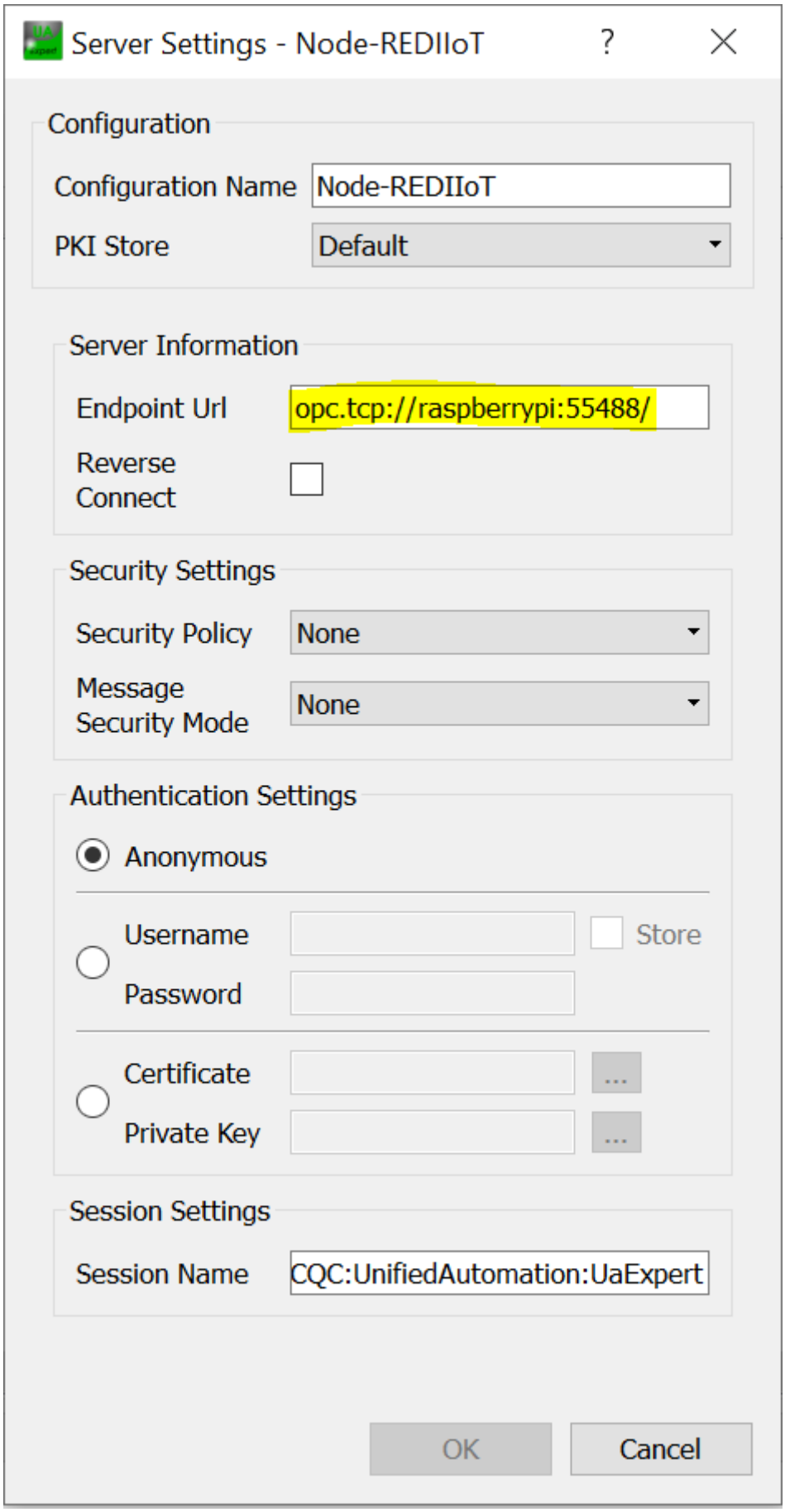

Add Server
Configuration Name
PKI Store
Default
Discovery
Advanced
Endpoint Filter:
No Filter
< Double click to Add Reverse Discovery... >
Custom Discovery
< Double click to Add Server... >
opc.tcp://127.0.0.1:53881
opc.tcp://192.168.1.64:55488
Node-RED (opc.tcp://raspberrypi:55488/)
None - None (uatcp-uasc-uabinary)
Basic128Rsa15 - Sign (uatcp-uasc-uabinary)
Basic256 - Sign (uatcp-uasc-uabinary)
Basic256Sha256 - Sign (uatcp-uasc-uabinary)
Aes128_Sha256_RsaOaep - Sign (uatcp-uasc-uabinary)
Basic128Rsa15 - Sign & Encrypt (uatcp-uasc-uabinary)
Basic256 - Sign & Encrypt (uatcp-uasc-uabinary)
Basic256Sha256 - Sign & Encrypt (uatcp-uasc-uabinary)
Authentication Settings
Anonymous
Username
Store
Password
Certificate
Private Key
Connect Automatically
OK
Cancel



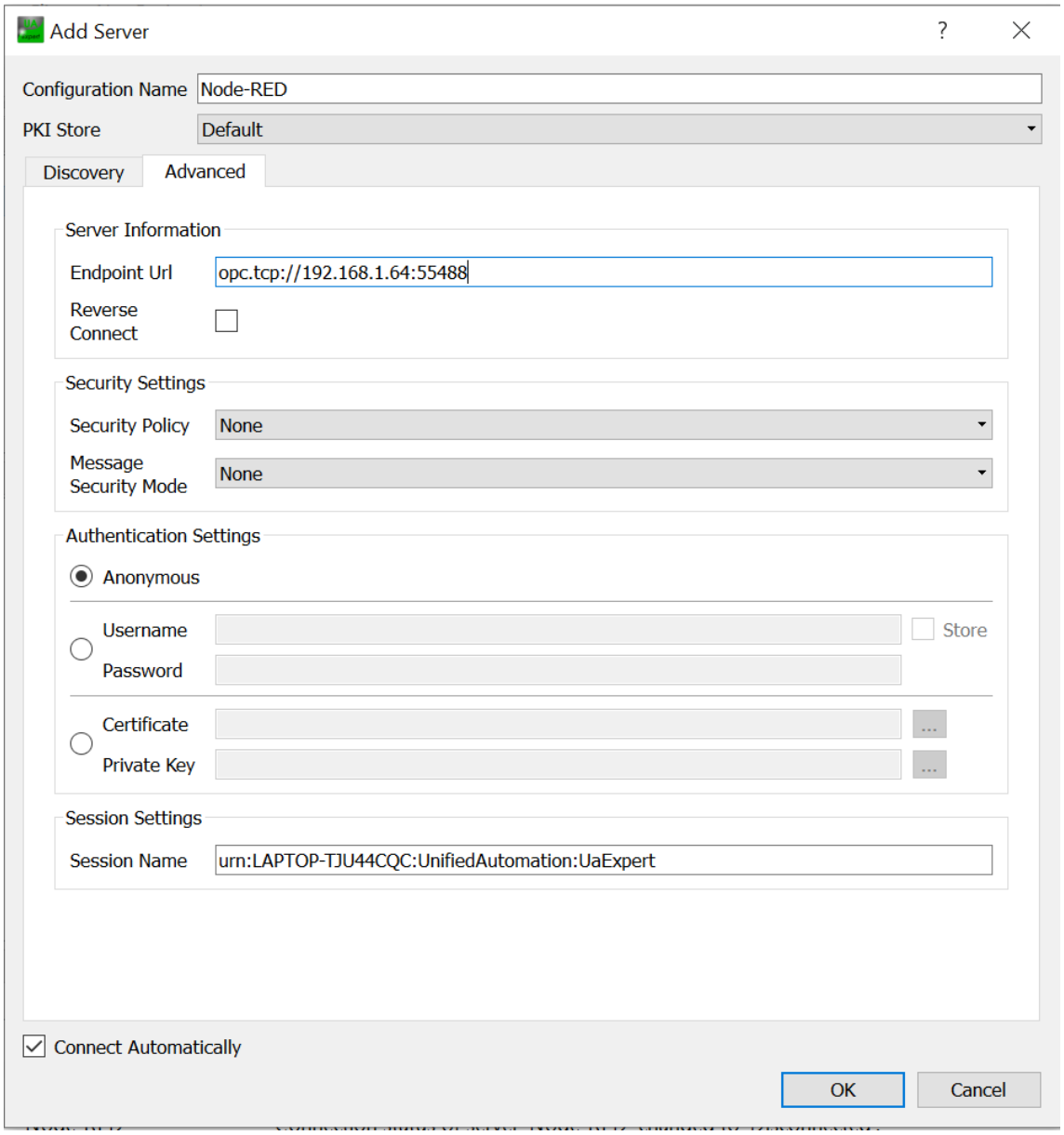
This way does not work

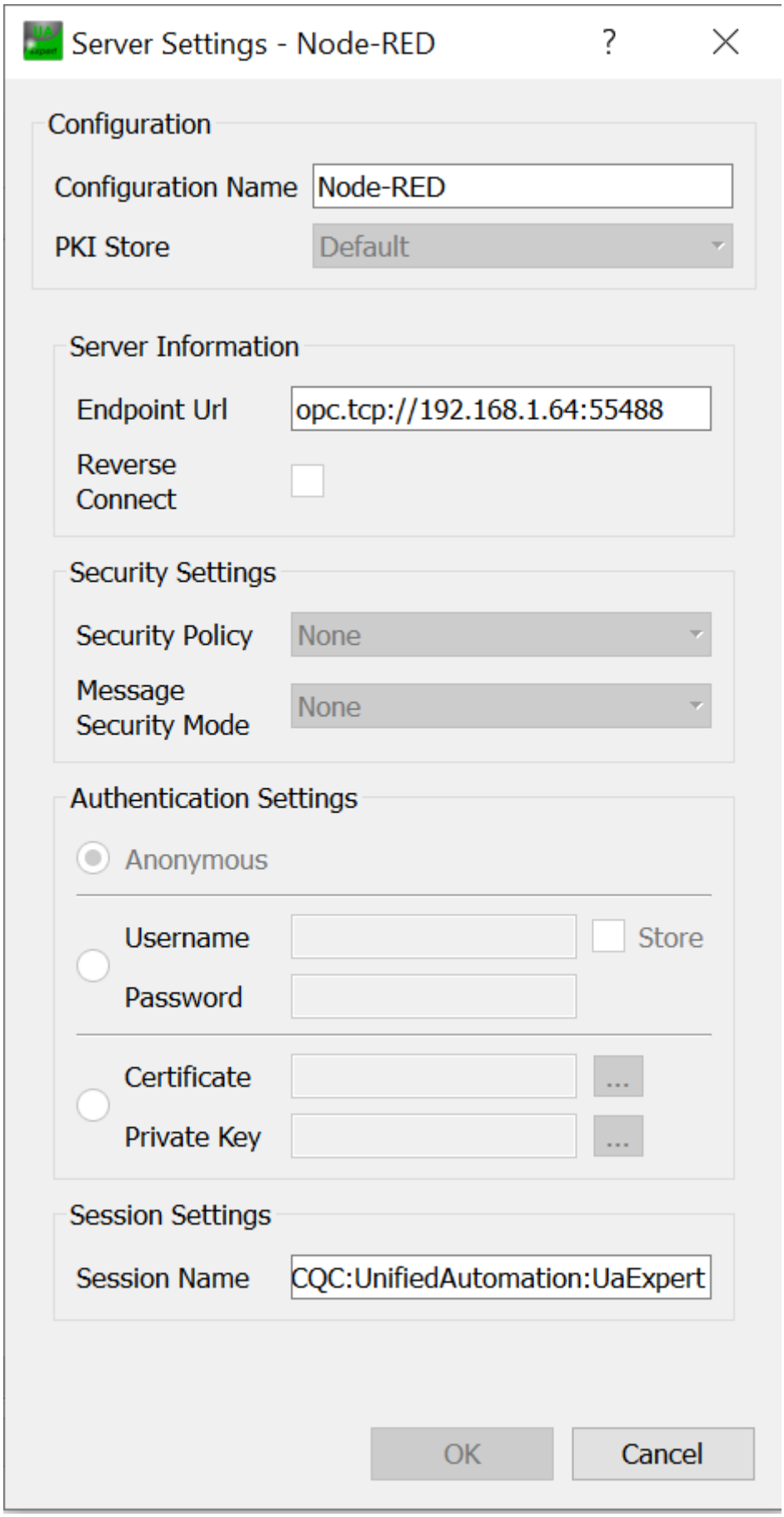


Since the url is not correct

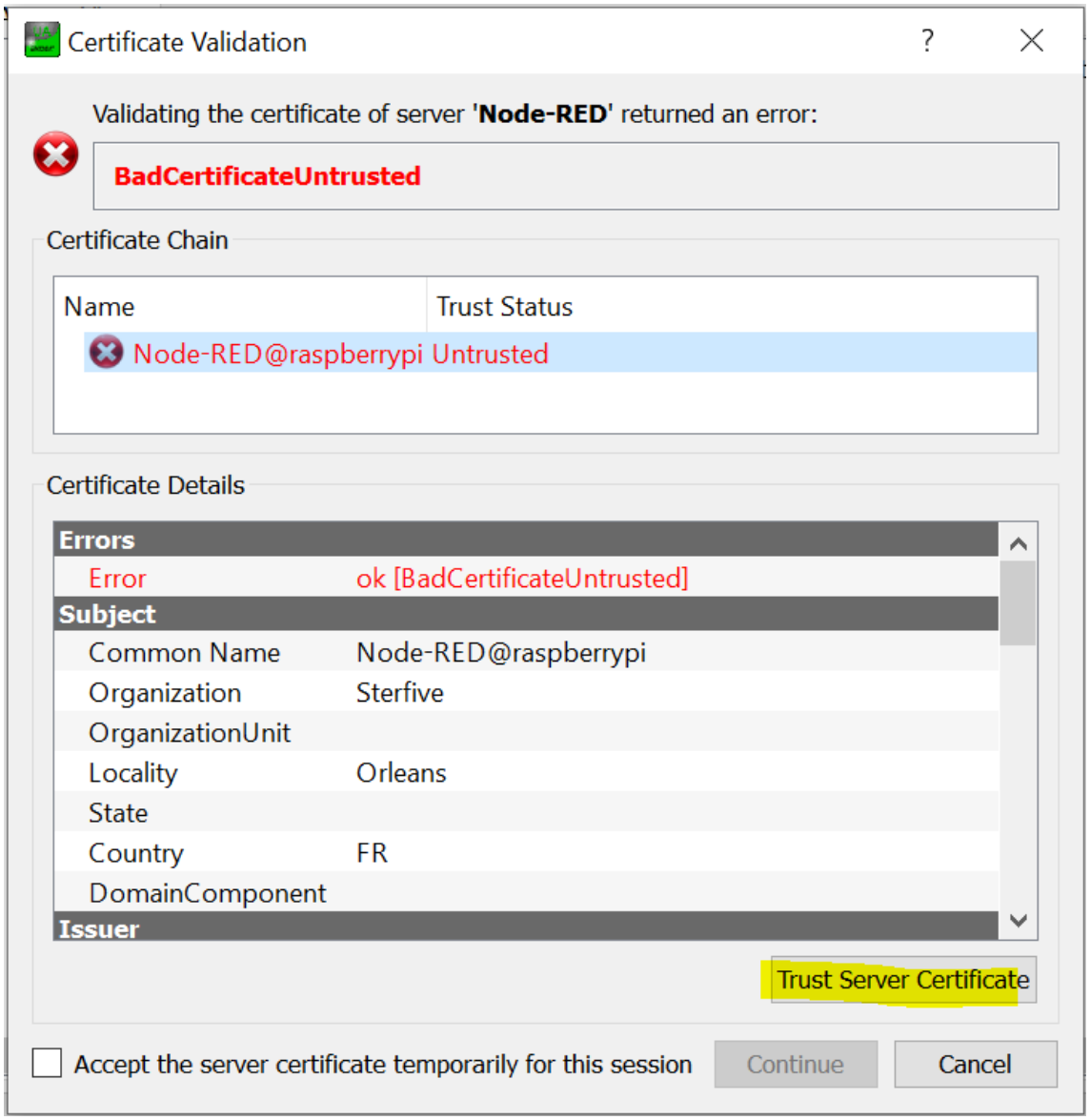

Server Settings - Node-REDIIoT
Configuration
Configuration Name
Node-REDIIoT
PKI Store
Default
Server Information
Endpoint Url
opc.tcp://raspberrypi:55488/
Reverse Connect
Security Settings
Security Policy
None
Message Security Mode
None
Authentication Settings
Anonymous
Username
Store
Password
Certificate
Private Key
Session Settings
Session Name
CQC:UnifiedAutomation:UaExpert
OK
Cancel

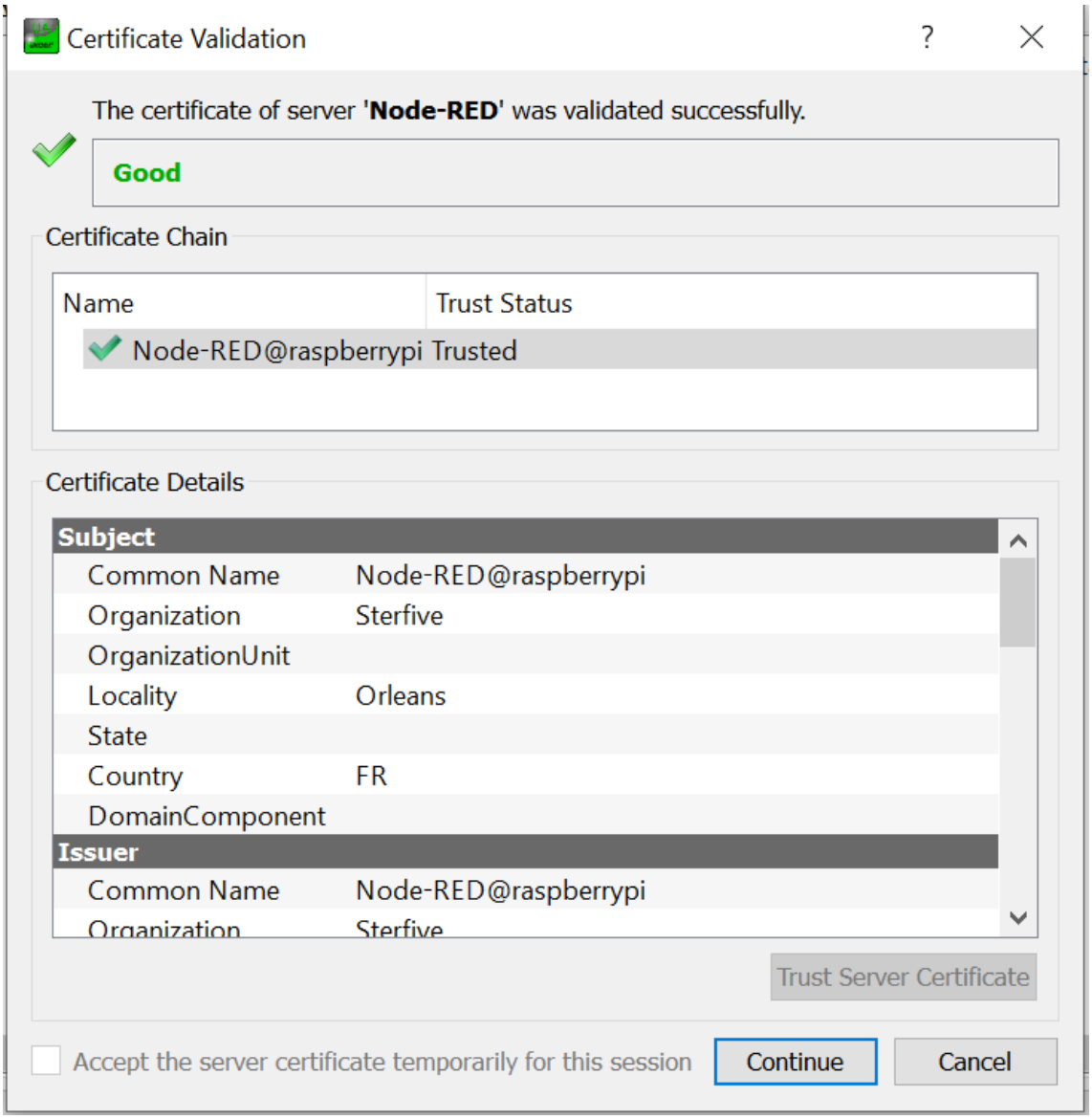
Let’s do it this way instead



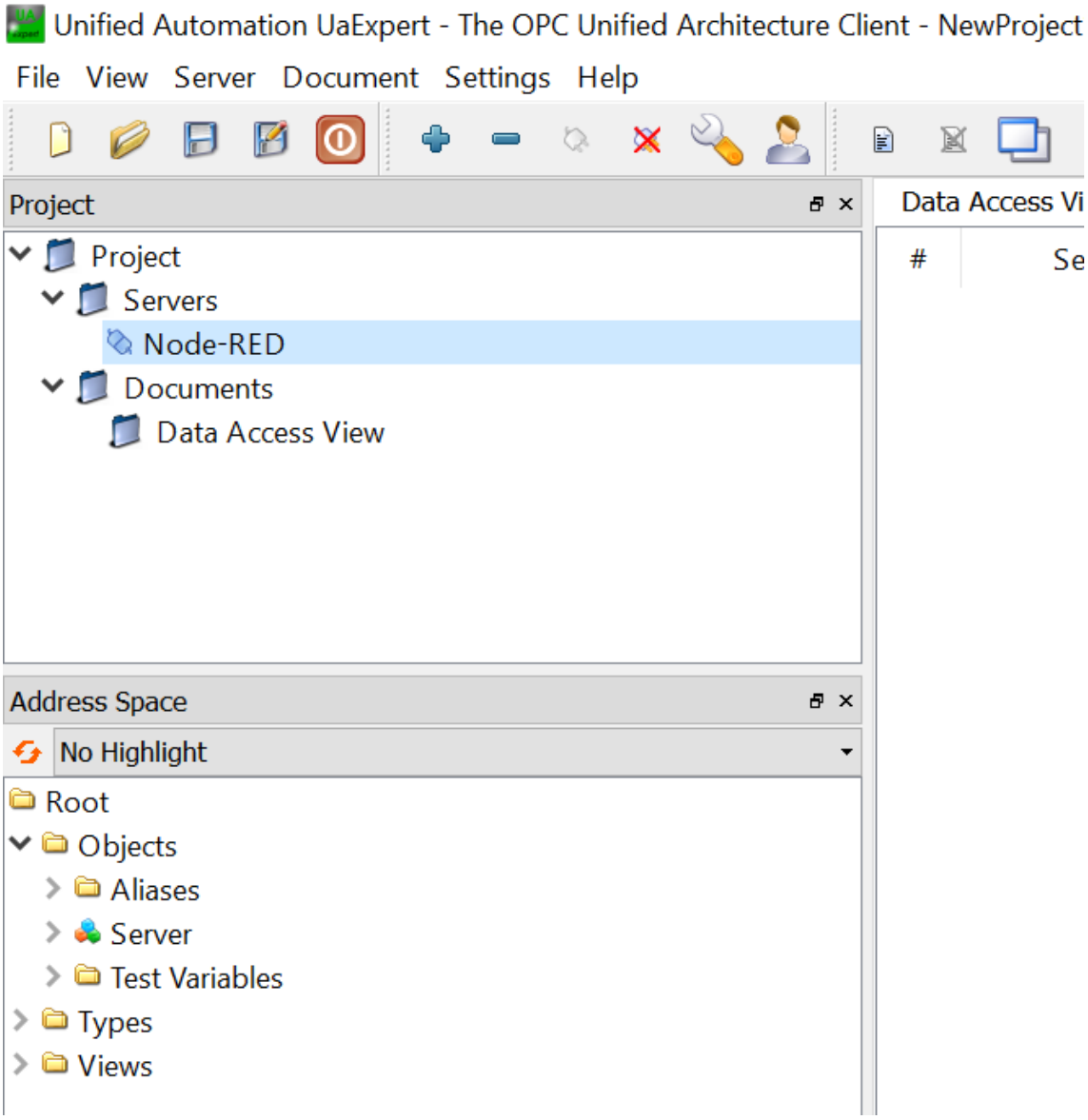

Server Settings - Node-RED
Configuration
Configuration Name
Node-RED
PKI Store
Default
Server Information
Endpoint Url
opc.tcp://192.168.1.64:55488
Reverse Connect
Security Settings
Security Policy
None
Message Security Mode
None
Authentication Settings
Anonymous
Username
Store
Password
Certificate
Private Key
Session Settings
Session Name
CQC:UnifiedAutomation:UaExpert
OK
Cancel

On first connection the client is rejected, let’s trust it

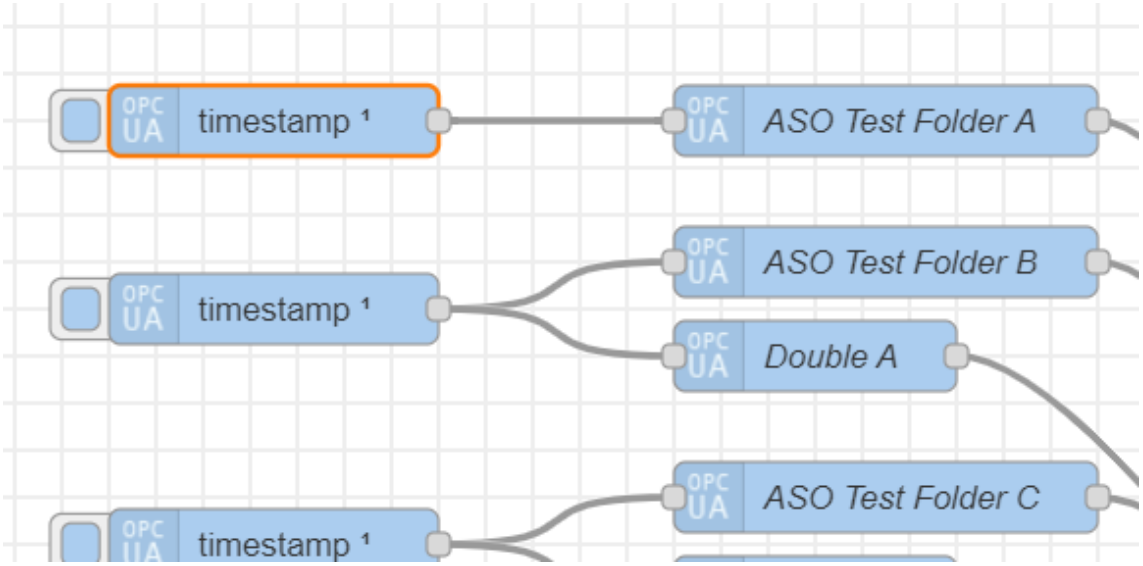


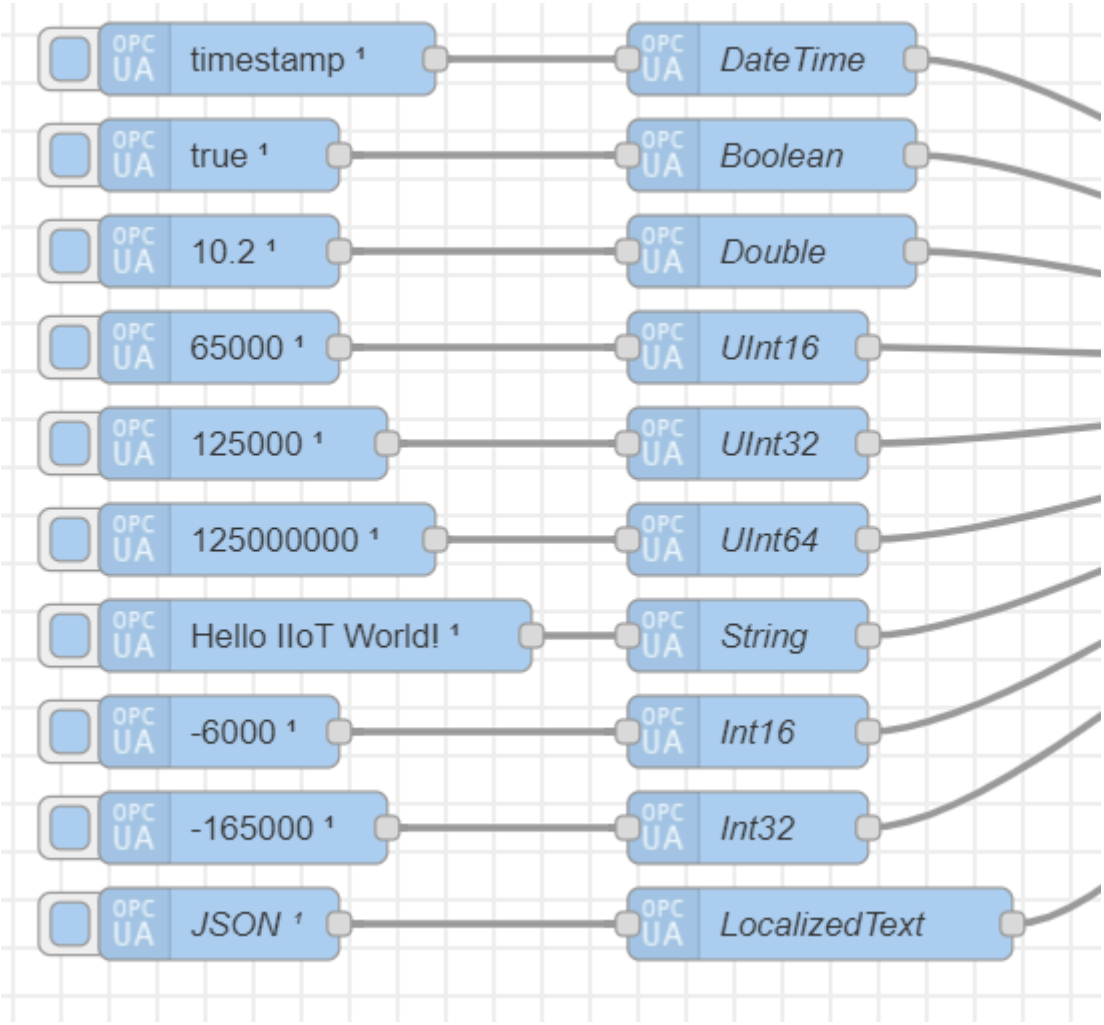


Now we can access the variables

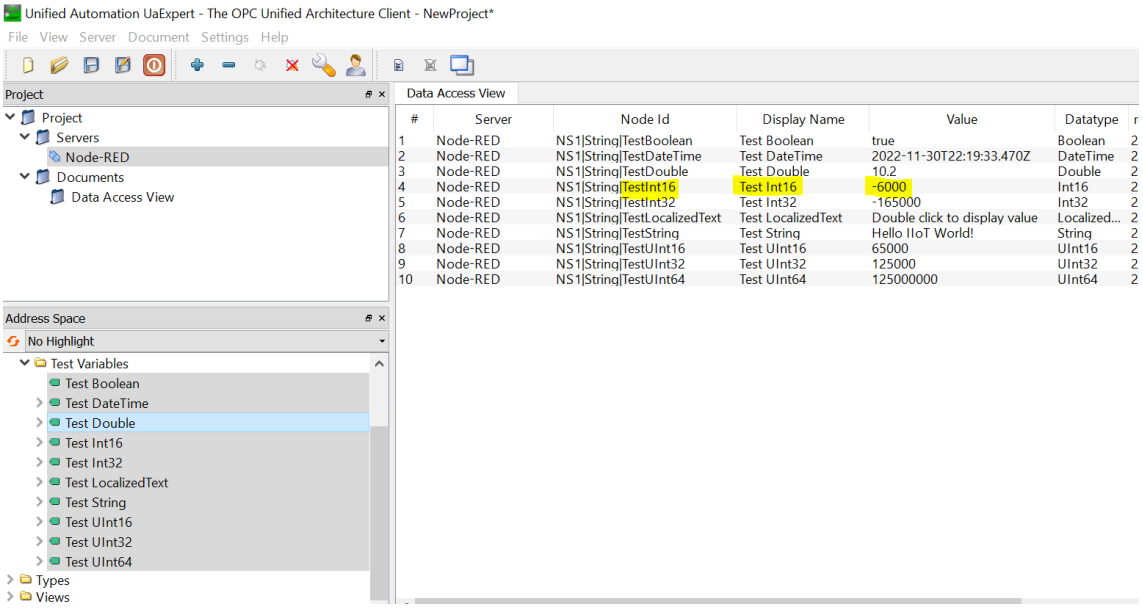


All the variables and folders are created at start



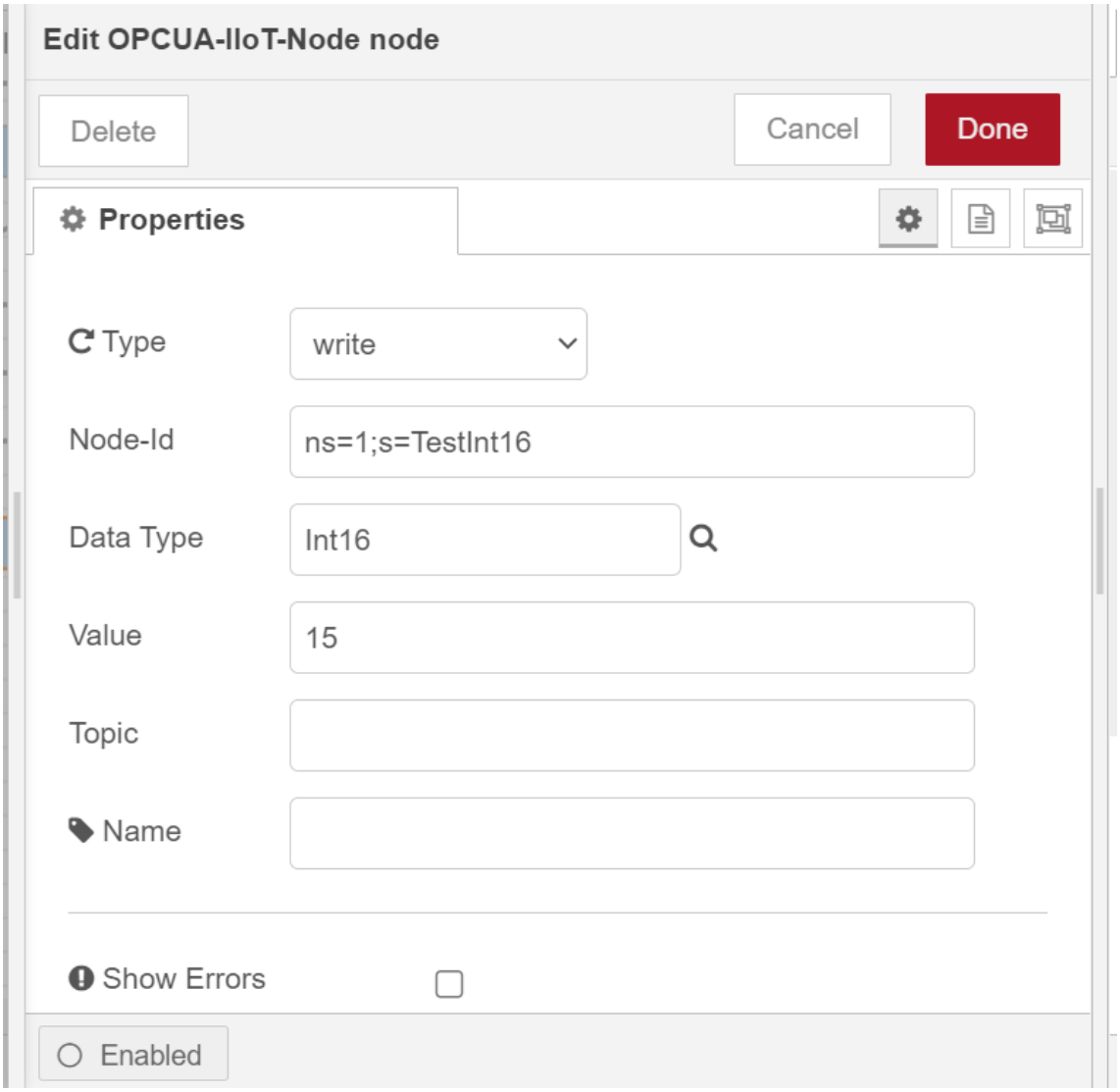


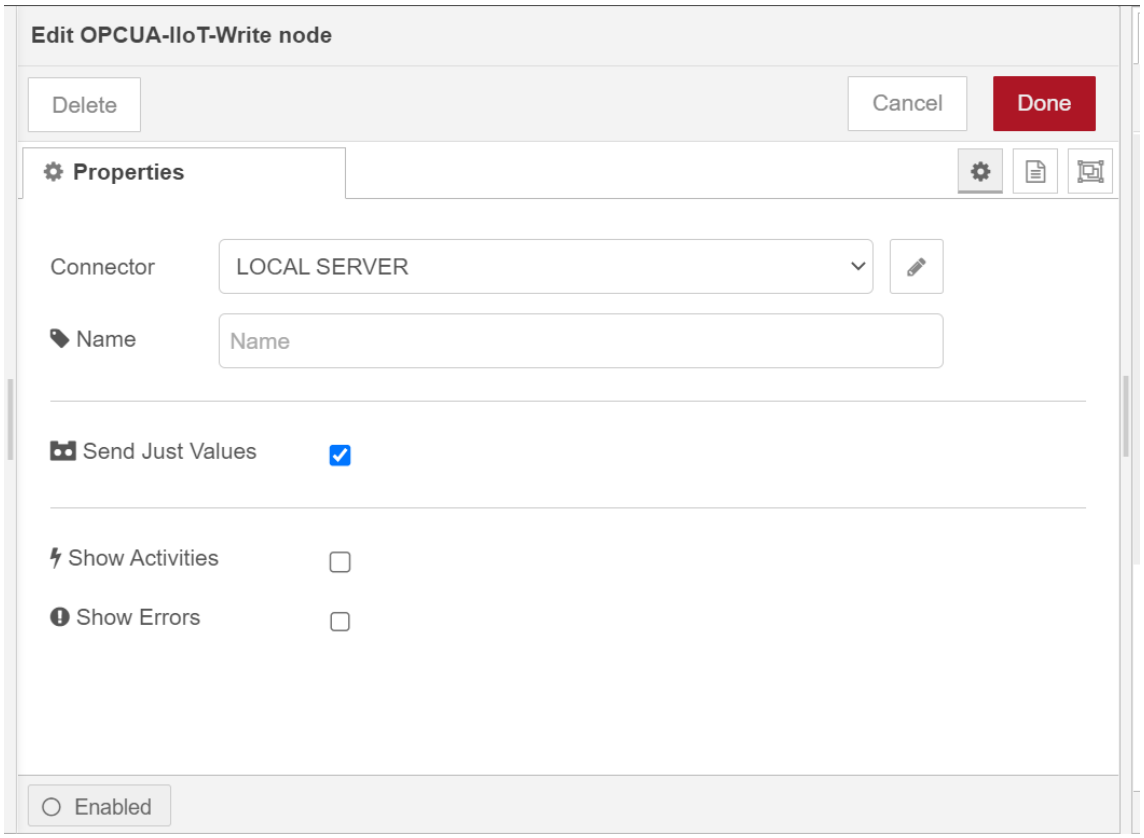
The variables are created but they do not change or update state.

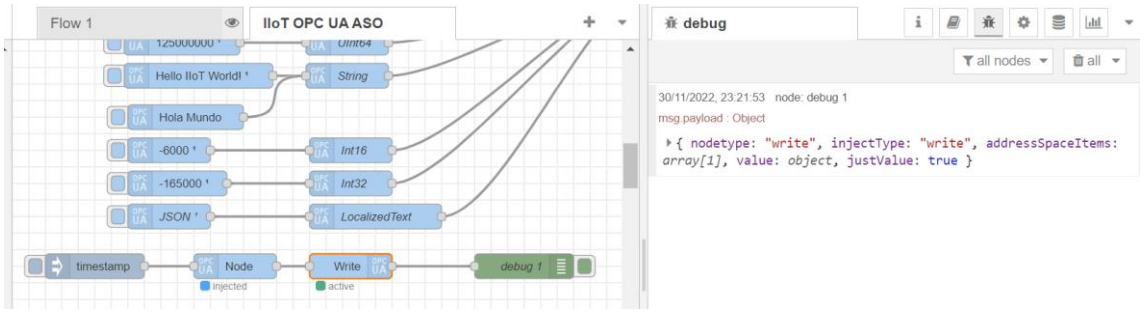


Let’s change the value of the integer16 variable


timestamp
OPC UA
Node
new
Write
OPC UA
active
debug 1

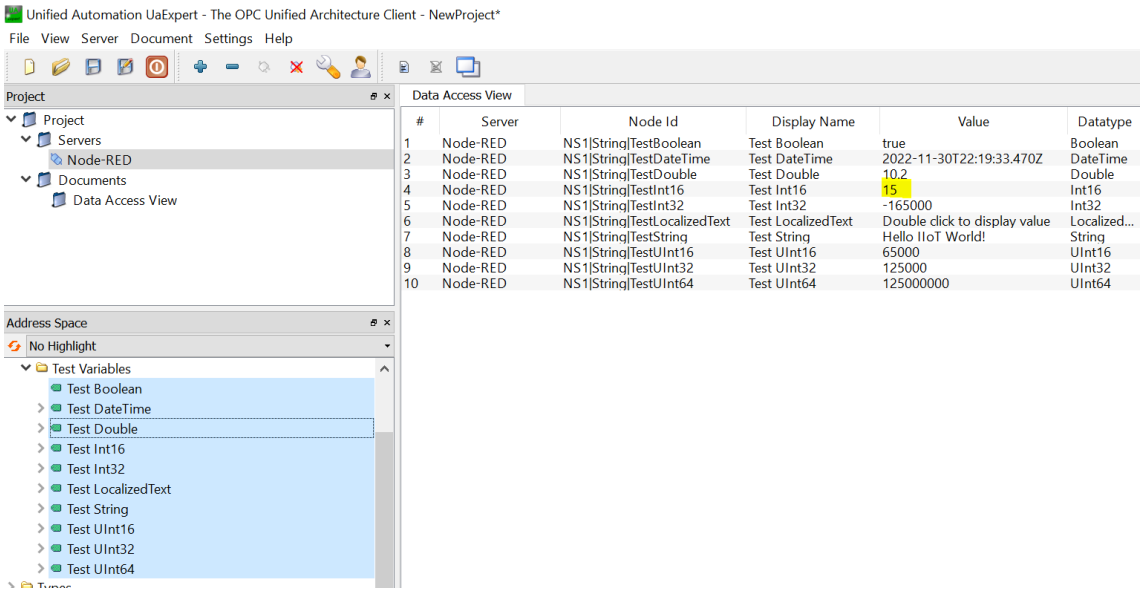

Edit OPCUA-IIoT-Node node
Delete
Cancel
Done
Properties
Type
write
Node-Id
ns=1;s=TestInt16
Data Type
Int16
Value
15
Topic
Name
Show Errors
Enabled

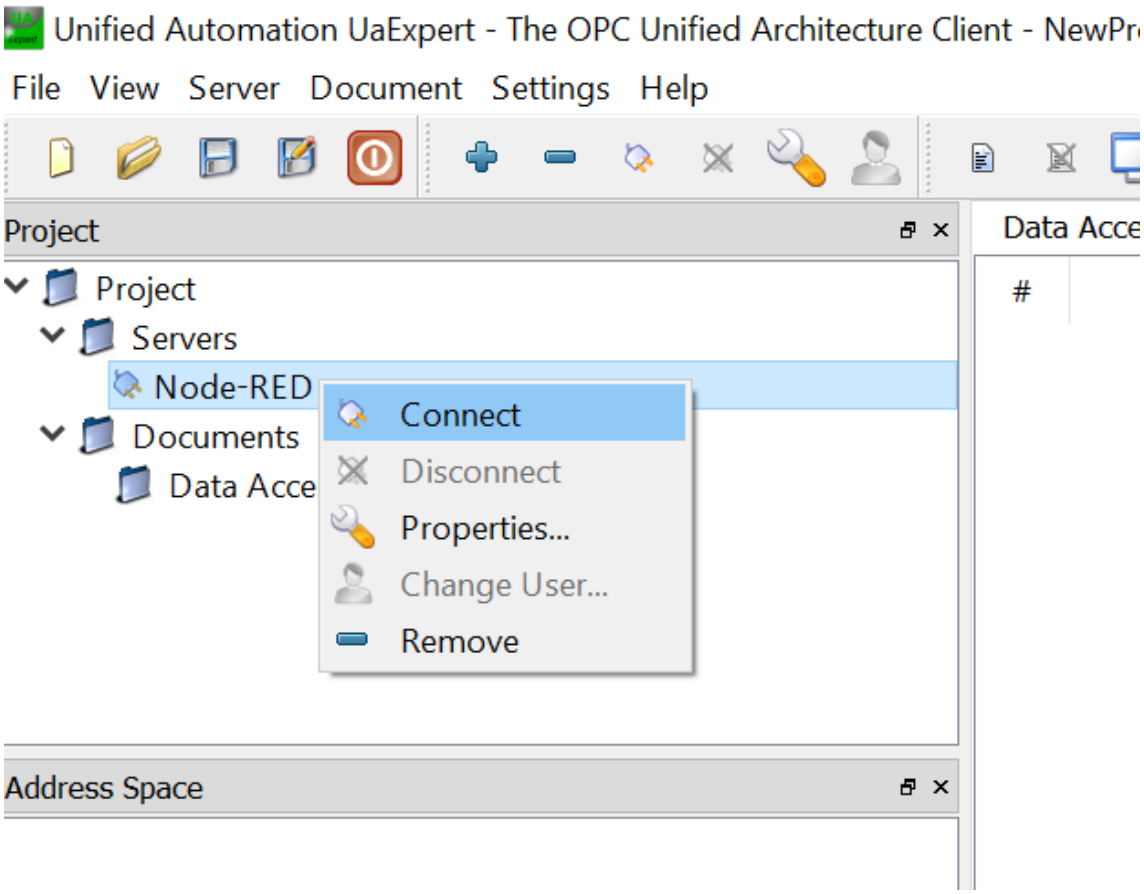




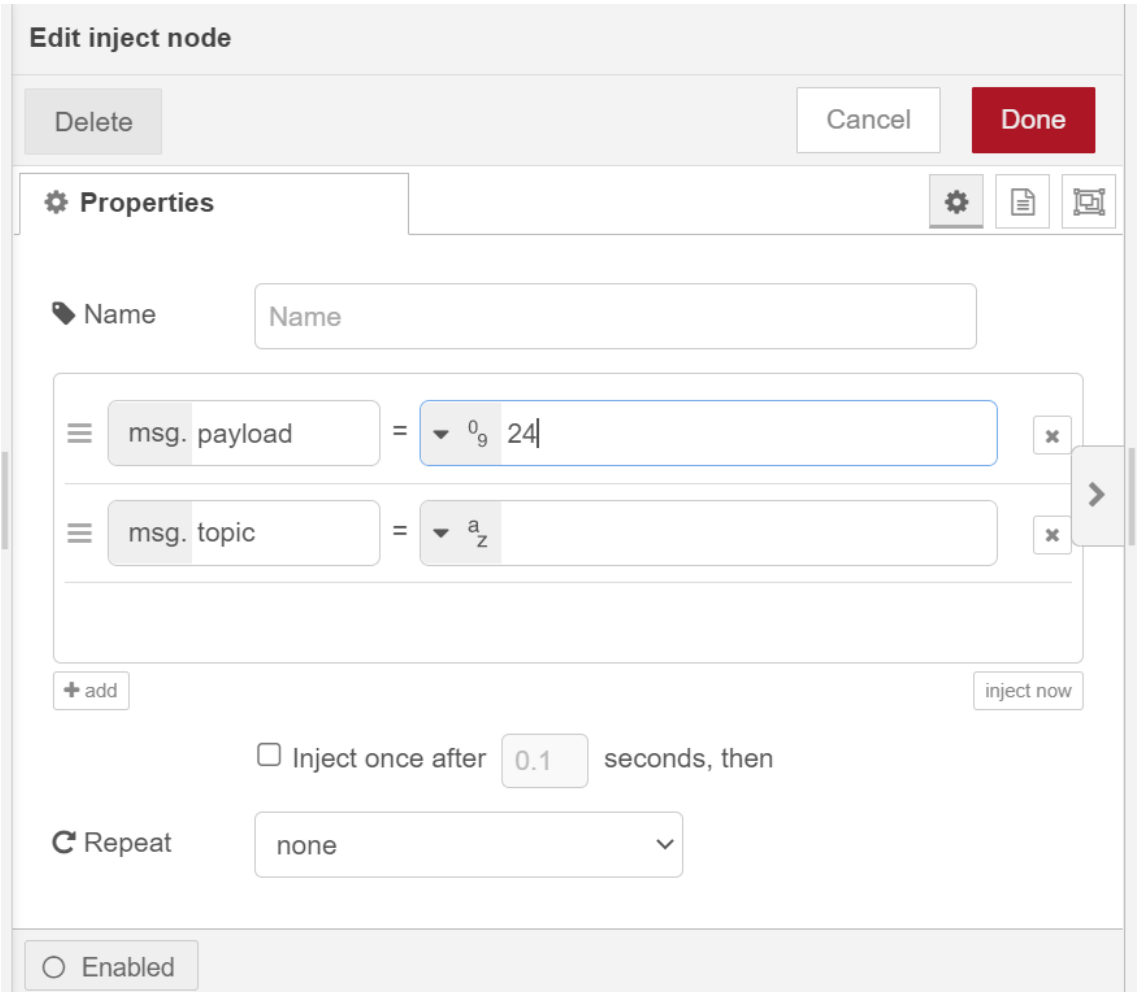
30/11/2022, 23:21:53 node: debug 1
msg.payload : Object

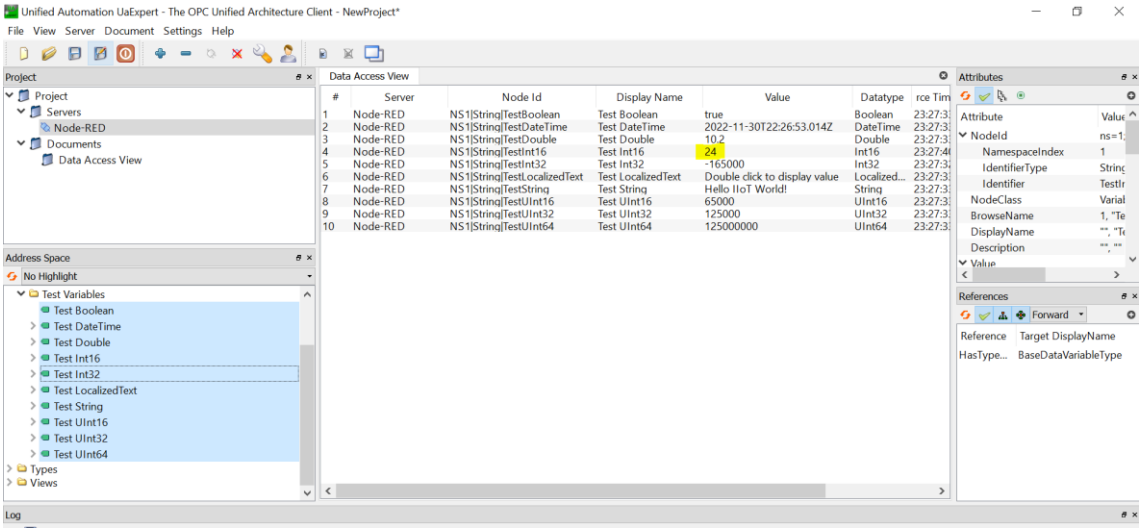

```
object
  nodetype: "write"
  injectType: "write"
addressSpaceItems: array[1]
  0: object
     name: ""
     nodeId: "ns=1;s=TestInt16"
     datatypeName: "Int16"
value: object
  statusCodes: array[1]
    0: object
       value: 0
  justValue: true
```

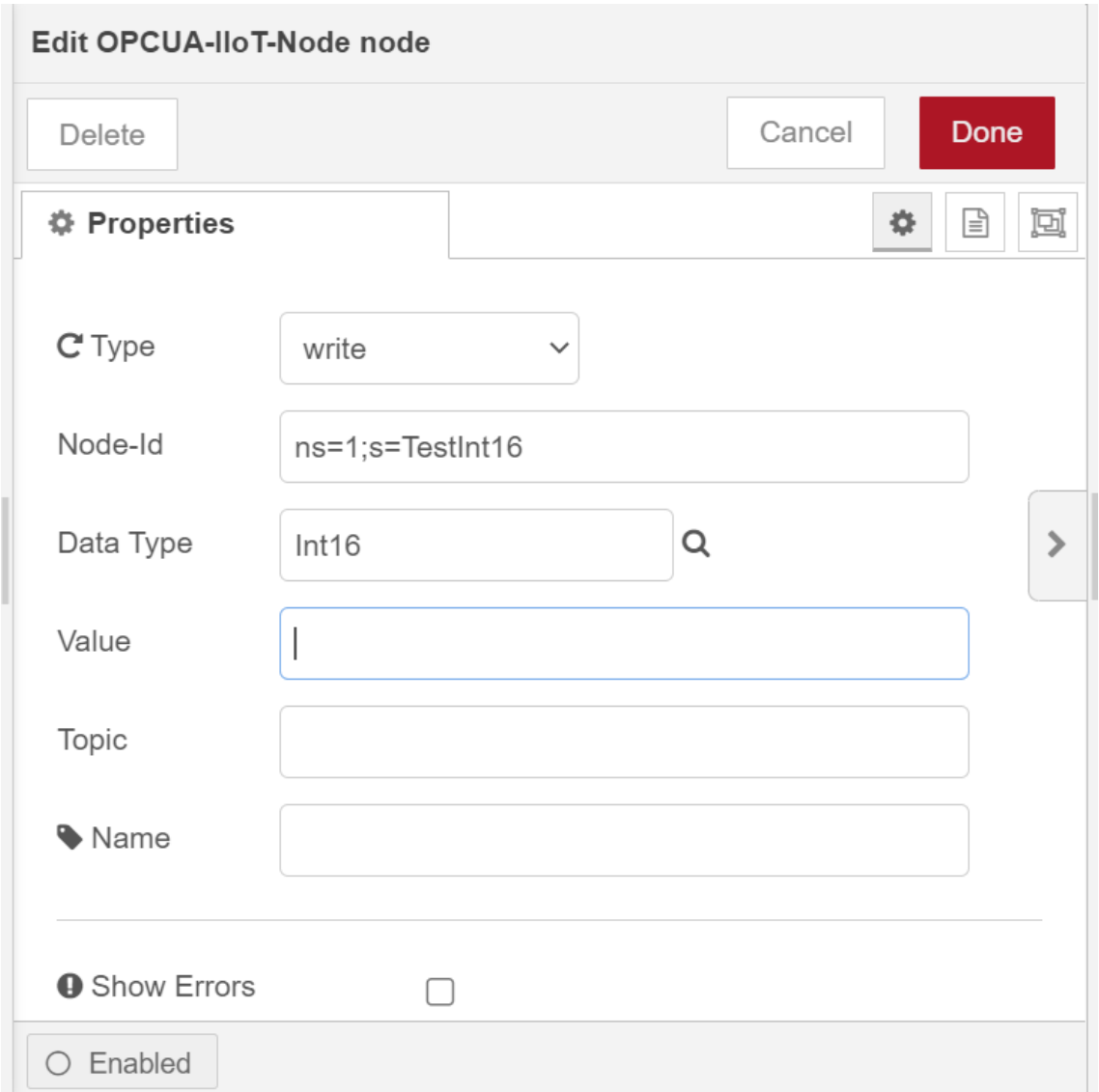




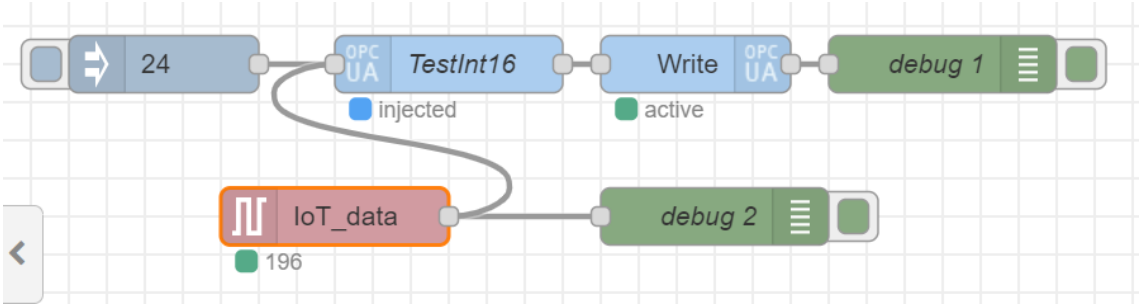
Let’s try to entre the value as a variable


Edit inject node
Delete
Cancel
Done
Properties
Name
Name
msg. payload
=
24
msg. topic
=
add
inject now
Inject once after
0.1
seconds, then
Repeat
none
Enabled

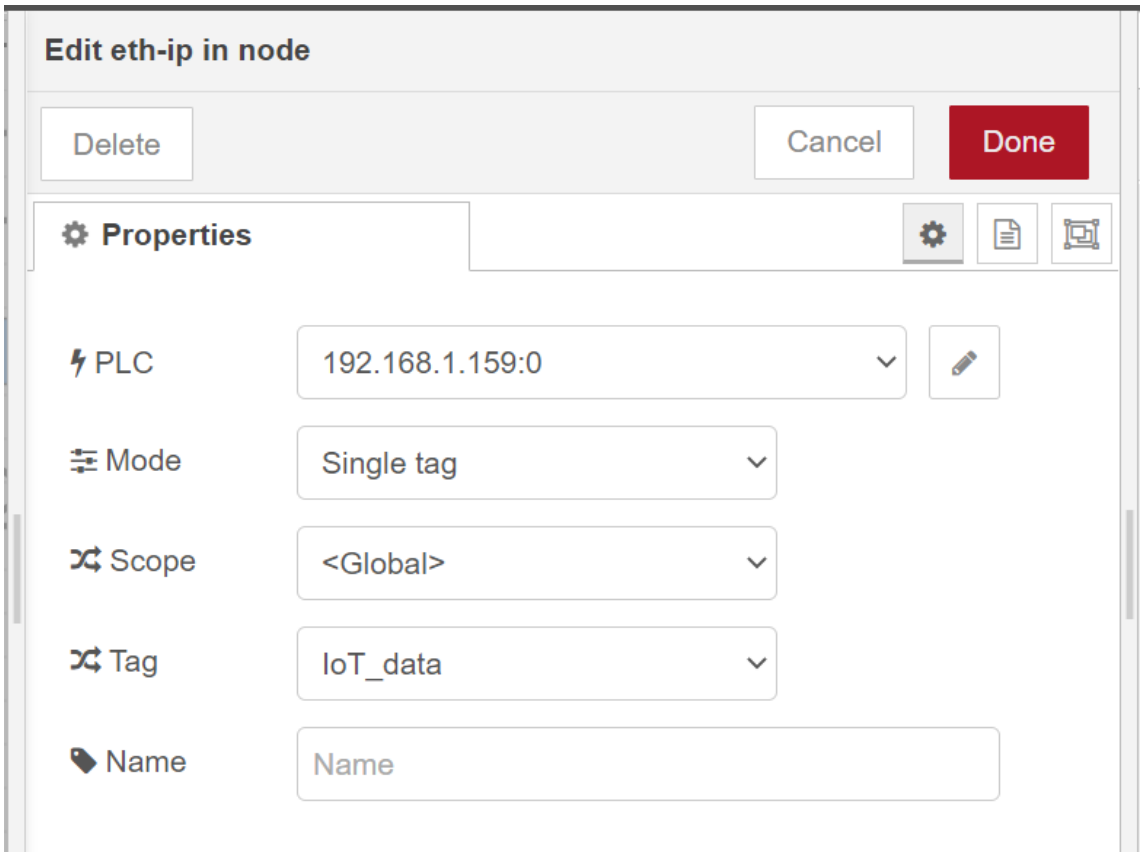

Unified Automation UaExpert - The OPC Unified Architecture Client - NewProject*
File View Server Document Settings Help
Project
Project
Servers
Node-RED
Documents
Data Access View
Address Space
No Highlight
Test Variables
Test Boolean
Test DateTime
Test Double
Test Int16
Test Int32
Test LocalizedText
Test String
Test UInt16
Test UInt32
Test UInt64
Types
Views
Data Access View
Server
Node Id
Display Name
Value
Datatype
Node-RED
NS1|String|TestBoolean
Test Boolean
true
Boolean
Node-RED
NS1|String|TestDateTime
Test DateTime
2022-11-30T22:26:53.014Z
DateTime
Node-RED
NS1|String|TestDouble
Test Double
10.2
Double
Node-RED
NS1|String|TestInt16
Test Int16
24
Int16
Node-RED
NS1|String|TestInt32
Test Int32
-165000
Int32
Node-RED
NS1|String|TestLocalizedText
Test LocalizedText
Double click to display value
Node-RED
NS1|String|TestString
Test String
Hello IIoT World!
String
Node-RED
NS1|String|TestUInt16
Test UInt16
65000
UInt16
Node-RED
NS1|String|TestUInt32
Test UInt32
125000
UInt32
Node-RED
NS1|String|TestUInt64
Test UInt64
125000000
UInt64
Attributes
Attribute
NodeId
NamespaceIndex
IdentifierType
Identifier
NodeClass
BrowseName
DisplayName
Description
Value
References
Forward
Reference
Target DisplayName
HasType...
BaseDataVariableType
Log



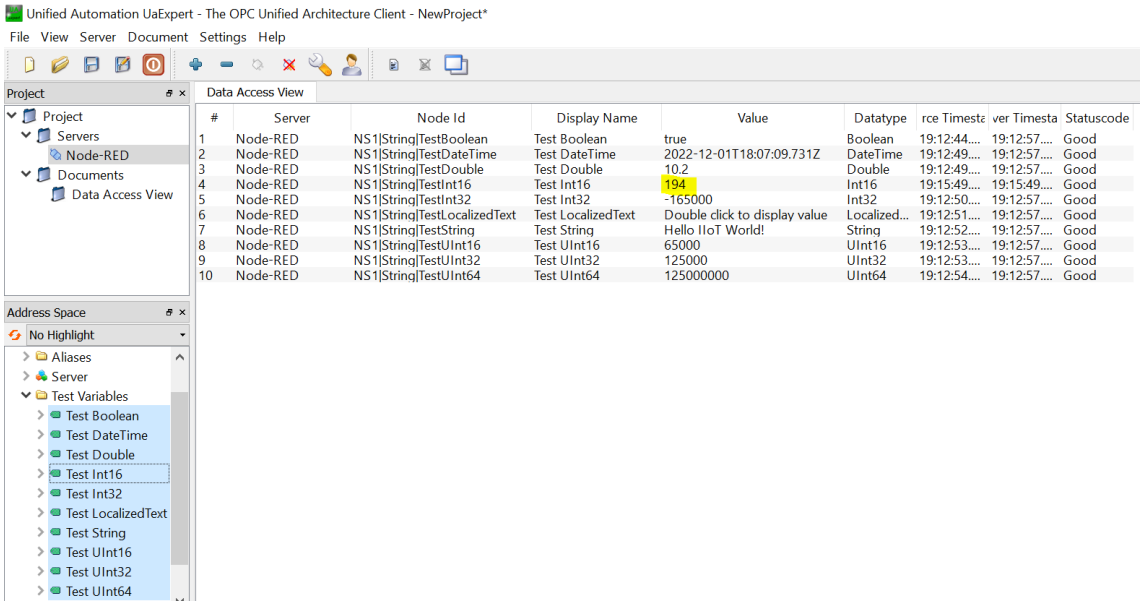
Now let's access the PLC data and inject to the OPC server



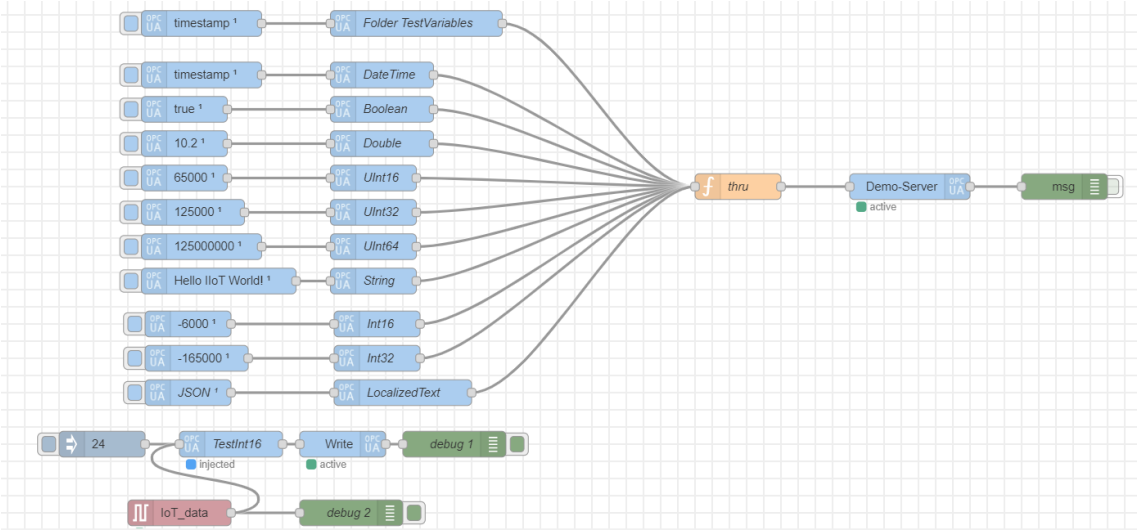
The PLC address is the IP address of your laptop if you are using FT Logix Echo



Now we can see the value being updated



Let’s erase the unneeded nodes from example



You can find the code here

https://github.com/xavierflorensa/EtherNet-IP-to-OPC-UA-server