

NR500 Series Industrial Cellular VPN Router

Application Note 064

Modbus LwM2M

Version: V1.0.0
Date: Aug 2022
Status: Confidential



Directory

1. Introduction	3
1.1 Overview	3
1.2 Compatibility	3
1.3 Version	3
1.4 Corrections	3
2. Topology	4
3. Configuration	5
3.1 LwM2M Server Configuration	5
3.2 Configuration on Modbus Slave	9
3.3 Configuration on Modbus Poll	9
3.4 Configuration on Modbus LwM2M	11
4. Testing	13

1. Introduction

1.1 Overview

This document contains information regarding the configuration and use of Modbus LwM2M.

This guide has been written for use by technically competent personnel with a good understanding of the communications technologies used in the product, and of the requirements for their specific application.

1.2 Compatibility

This application note applies to:

Models Shown: NR500 series.

Firmware Version: V1.1.7(3b5122d) or newer

Other Compatible Models: None

1.3 Version

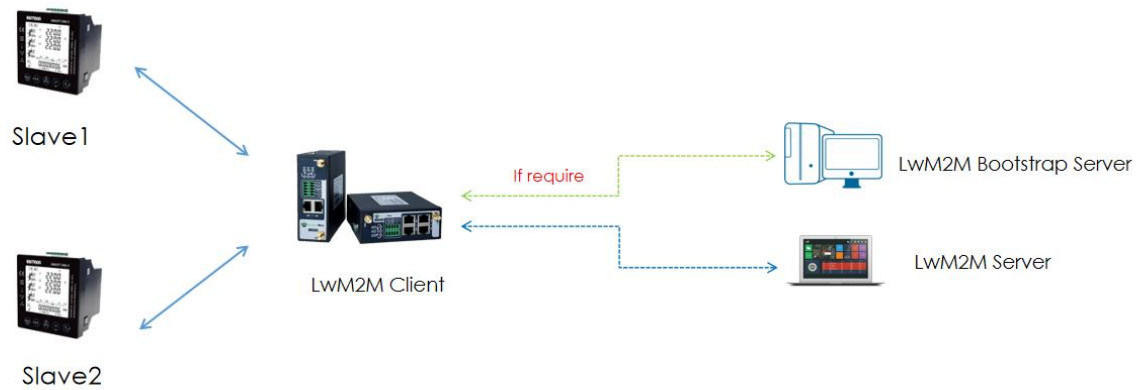
Updates between document versions are cumulative. Therefore, the latest document will include all the content of previous versions.

Release Date	Doc. Version	Firmware Version	Change Description
2022/08/24	V1.0.0	V1.1.7(3b5122d)	First released

1.4 Corrections

Appreciate for corrections or rectifications to this application note, and if any request for new application notes please email to: support@navigateworx.com

2. Topology



1. NR500 Router act as Modbus Master and LwM2M Client, and connect to Modbus Slave via Ethernet, RS232 or RS485 interface.
2. NR500 Router poll the modbus data from modbus slave and send to the LwM2M Bootstrap Server or LwM2M Server via LwM2M protocol.
3. LwM2M Bootstrap Server or LwM2M Server can write the register value to Modbus Slave.

3. Configuration

3.1 Lwm2m Server Configuration

1. Setup Lwm2m Server by Eclipse Leshan project, which requires JAVA running environment. Please Prepare JDK environment before the testing. Our testing environment as below:

```
E:\leshan>java -version
openjdk version "1.8.0_302"
OpenJDK Runtime Environment (Temurin)(build 1.8.0_302-b08)
OpenJDK 64-Bit Server VM (Temurin)(build 25.302-b08, mixed mode)
E:\leshan>
```

2. Add a "modules" folder, and create **.XML** file to create instances and object resources.

Modbus-To-Lwm2m > leshan > modules			
名称	修改日期	类型	大小
31024.xml	2022/8/23 16:44	XML 文档	3 KB

The configuration of **31024.xml** like below:

```
<?xml version="1.0" encoding="UTF-8"?>
<LWM2M xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://openmobilealliance.org/tech/profiles/LWM2M-v1_1.xsd">
  <Object ObjectType="MODefinition">
    <Name>Test Object</Name>
    <Description1>Tommy Test object.</Description1>
    <ObjectID>31024</ObjectID>
    <ObjectURN>urn:oma:lwm2m:x:31024</ObjectURN>
    <LWM2MVersion>1.1</LWM2MVersion>
    <ObjectVersion>1.0</ObjectVersion>
    <MultipleInstances>Multiple</MultipleInstances>
    <Mandatory>Optional</Mandatory>
    <Resources>
      <Item ID="1">
        <Name>boolean 1</Name>
        <Operations>RW</Operations>
        <MultipleInstances>Single</MultipleInstances>
        <Mandatory>Optional</Mandatory>
        <Type>Boolean</Type>
        <RangeEnumeration></RangeEnumeration>
```

```
<Units></Units>
  <Description>boolean value</Description>
</Item>
<Item ID="2">
  <Name>boolean 2</Name>
  <Operations>RW</Operations>
  <MultipleInstances>Single</MultipleInstances>
  <Mandatory>Optional</Mandatory>
  <Type>Boolean</Type>
  <RangeEnumeration></RangeEnumeration>
  <Units></Units>
  <Description>boolean value</Description>
</Item>
<Item ID="3">
  <Name>int16</Name>
  <Operations>RW</Operations>
  <MultipleInstances>Single</MultipleInstances>
  <Mandatory>Optional</Mandatory>
  <Type>Integer</Type>
  <RangeEnumeration></RangeEnumeration>
  <Units></Units>
  <Description>int 16 value</Description>
</Item>
<Item ID="4">
  <Name>uint16</Name>
  <Operations>RW</Operations>
  <MultipleInstances>Single</MultipleInstances>
  <Mandatory>Optional</Mandatory>
  <Type>Unsigned Integer</Type>
  <RangeEnumeration></RangeEnumeration>
  <Units></Units>
  <Description>uint 16 value</Description>
</Item>
<Item ID="5">
  <Name>int32</Name>
  <Operations>RW</Operations>
  <MultipleInstances>Single</MultipleInstances>
  <Mandatory>Optional</Mandatory>
  <Type>Integer</Type>
  <RangeEnumeration></RangeEnumeration>
  <Units></Units>
  <Description>int 32 value</Description>
</Item>
<Item ID="6">
```

```

<Name>float</Name>
<Operations>RW</Operations>
<MultipleInstances>Single</MultipleInstances>
<Mandatory>Optional</Mandatory>
<Type>Float</Type>
<RangeEnumeration></RangeEnumeration>
<Units></Units>
<Description>float value</Description>
</Item>
<Item ID="7">
  <Name>double</Name>
  <Operations>RW</Operations>
  <MultipleInstances>Single</MultipleInstances>
  <Mandatory>Optional</Mandatory>
  <Type>Float</Type>
  <RangeEnumeration></RangeEnumeration>
  <Units></Units>
  <Description>double value</Description>
</Item>
</Resources>
<Description2></Description2>
</Object>
</LWM2M>

```

3. Install Lwm2m Server or Lwm2m Bootstrap Server, here we test with Lwm2m Server.

Modbus-To-Lwm2M > leshan

名称	修改日期	类型	大小
data	2022/8/23 16:09	文件夹	
modules	2022/8/23 16:09	文件夹	
Californium3.bserver.properties	2022/4/29 9:25	PROPERTIES 文件	13 KB
Californium3.server.properties	2022/4/29 9:23	PROPERTIES 文件	13 KB
leshan-bserver-demo.jar	2022/2/25 5:06	JAR 文件	12,095 KB
leshan-client-demo.jar	2022/2/25 5:06	JAR 文件	6,628 KB
leshan-server-demo.jar	2022/2/25 5:06	JAR 文件	13,729 KB

4. Run leshan-server-demo.jar file with port 5683(CoAP) or port 5684(CoAPS) in CMD, the command as below:

java -jar leshan-server-demo.jar -wp 8090 -m modules

Note: a) Run the server demo with webserver port 8090, because the default port 8080 will conflict when the Bootstrap server run.

b) If Run Lwm2m Bootstrap Server with port 6683(CoAP) or port 6684(CoAPS) in CMD, the command: **java -jar leshan-bserver-demo.jar -lp 6683 -slp 6684**

```

C:\Windows\system32\cmd.exe - java -jar leshan-server-demo.jar -wp 8090 -m modules
E:\leshan>dir
驱动器 E 中的卷是 HD备份磁盘
卷的序列号是 0D07-CAC3

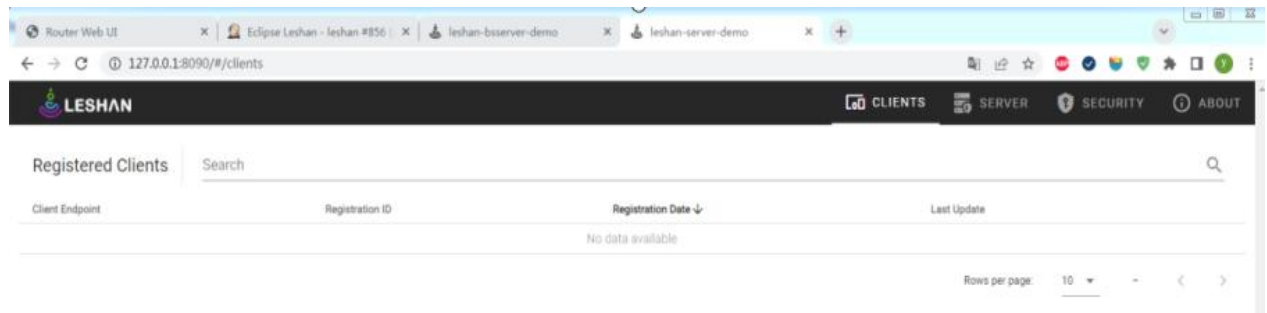
E:\leshan 的目录

2022/04/26 17:12 <DIR>      .
2022/04/26 17:12 <DIR>      ..
2022/04/21 11:48             13,620 Californium3.bsserver.properties
2022/03/30 15:50             13,474 Californium3.client.properties
2022/03/30 10:16             13,607 Californium3.server.properties
2022/04/21 11:49 <DIR>      data
2022/02/25 05:06           12,385,207 leshan-bsserver-demo.jar
2022/02/25 05:06             6,786,236 leshan-client-demo.jar
2022/02/25 05:06           14,058,358 leshan-server-demo.jar
2022/04/26 17:12 <DIR>      modules
                6 个文件      33,270,502 字节
                4 个目录      241,350,184,960 可用字节

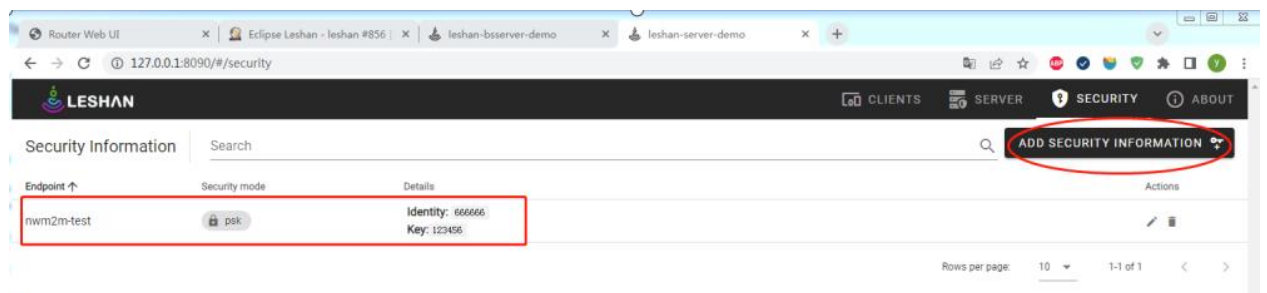
E:\leshan>java -jar leshan-server-demo.jar -wp 8090 -m modules
2022-04-26 18:04:58,364 LeshanServer [INFO] LWM2M server started at coap://0.0.0.0/0.0.0.0:5683 coaps://0.0.0.0/0.0.0.0:5684
2022-04-26 18:04:58,447 LeshanServerDemo [INFO] Web server started at http://0.0.0.0:8090/.
  
```

Running successfully.

5. Then access the server via <http://127.0.0.1:8090>. ([Lwm2M Bootstrap Server http://127.0.0.1:8080](http://127.0.0.1:8080).)

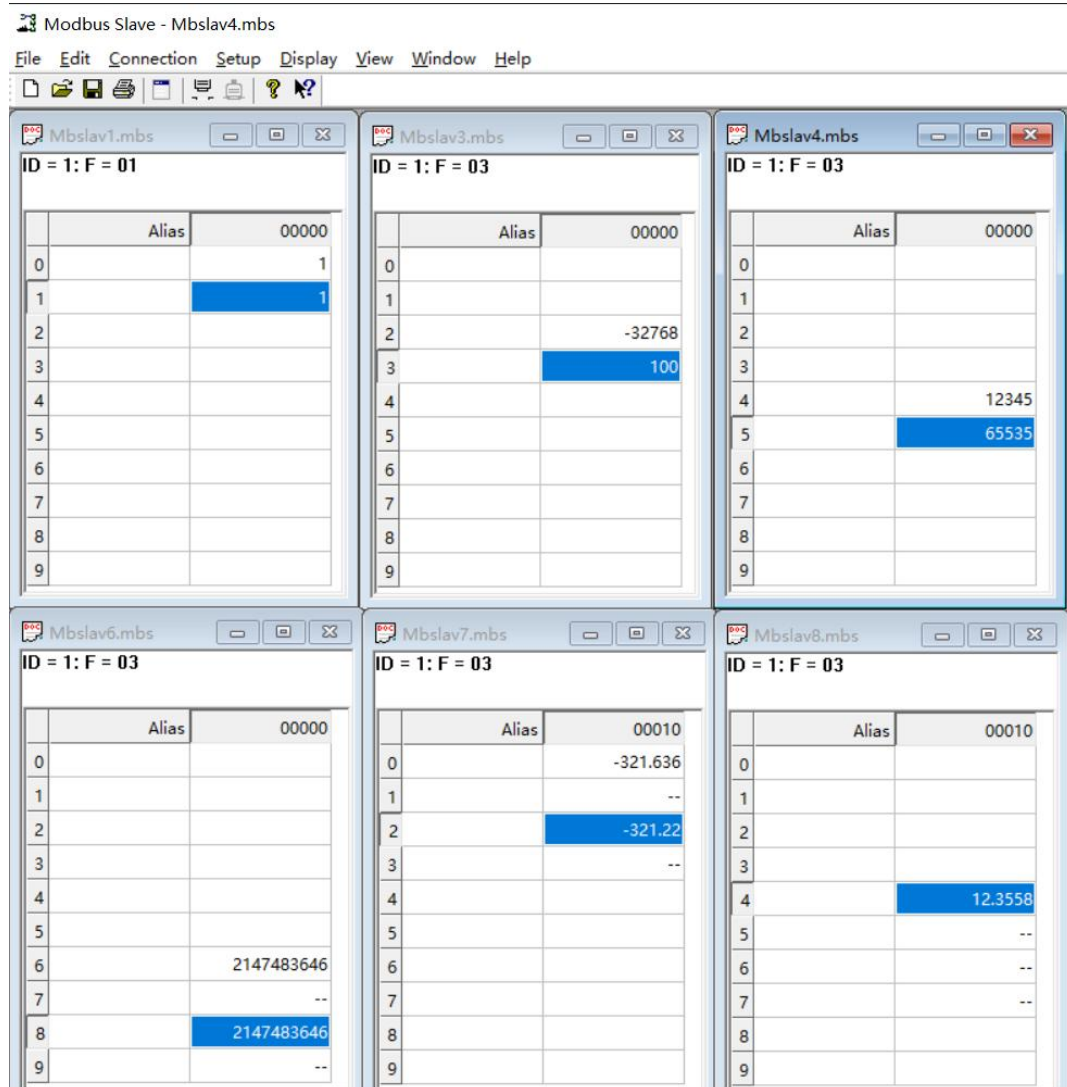


6. Add the Security information on Lwm2M Server, please make sure that match Lwm2M Client configuration. Add Endpoint Name, select Security mode, and add the PSK Identity and PSK Key.



3.2 Configuration on Modbus Slave

1. Here we use “Modbus Slave” software to simulate the end device (Modbus Slave device), and the **TCP Port: 1002, Slave ID: 1**, like below setting:



3.3 Configuration on Modbus Poll

1. Go to **Application>Modbus Master>Modbus Poll**, add a “Connection List” and specify the “Connection Type” as “TCP”, specify the “TCP Setting” to connect to Modbus Slave, like below:

Connection Settings

Connection List

Index:
 Enable:
 Description:
 Scan Rate: ?
 Reconnect Interval: ?
 Response Timeout: ?
 Delay Between Polls: ?
 Connection Type: v
 Enable Show Status:
 Enable Verbose Log:

TCP Settings

Server Address:
 Server Port:
 Connection Timeout: ?

Channel List

Index	Enable	Description	Slave ID	Function Code	Register Address	+
1	true	boolean	1	01-Coil-Status	0	✎ ✕

2. Click Save>Apply.
3. Enable "Channel List", and specify the **Slave ID** as " 1 ", **Function Code** according slave1, **Register Address** like below, then it will poll the value from register specially address of Modbus Slave:

Channel List

Index	Enable	Description	Slave ID	Function Code	Register Address	+
1	true	boolean	1	01-Coil-Status	0	✎ ✕
2	true		1	03-Holding-Register	2	✎ ✕
3	true		1	03-Holding-Register	4	✎ ✕
4	true		1	03-Holding-Register	6	✎ ✕
5	true		1	03-Holding-Register	10	✎ ✕
6	true		1	03-Holding-Register	14	✎ ✕

4. Click Save>Save>Apply. *(Note: This is a secondary list, it needs to double click save)*

5. Go to **Application>Modbus Master>Status**, then we can check the router had read the value from Modbus Slave successfully.

Status								
Channel Status								
Index	Description	Connection Index	Type	Slave ID	Register Address	Function Code	Status	Value
1	boolean	1	TCP	1	0	1	Read successfully	1, 1
2		1	TCP	1	2	3	Read successfully	-32768, 100
3		1	TCP	1	4	3	Read successfully	12345
4		1	TCP	1	6	3	Read successfully	2147483646
5		1	TCP	1	10	3	Read successfully	-321.636353
6		1	TCP	1	14	3	Read successfully	12.355845
7		2	TCP	2	0	1	Read successfully	1
8		2	TCP	2	2	3	Read successfully	2
9		2	TCP	2	4	3	Read successfully	3
10		2	TCP	2	6	3	Read successfully	-2147483646
11		2	TCP	2	10	3	Read successfully	0.123456
12		2	TCP	2	14	3	Read successfully	22.334456

3.4 Configuration on Modbus LwM2M

1. Go to **Application>Modbus LwM2M>Modbus to LwM2M**, enable "Connection List", and specify Server address and Server port and Endpoint Name, Endpoint Name and PSK Identity and PSK key should be match the settings on LwM2M Server.

LwM2M Connection Settings

Connection Settings

Index:

Enable:

Endpoint Name:

Local Port:

Server Address:

Server Port:

Auth Type:

PSK Identity:

PSK Key:

Requested Bootstrap:

Lifetime:

CoAP Block Size:

Enable Verbose Log:

Modbus To LwM2M Mapping Channel

Index	Slave ID	Register Address	Object ID	Object Resource Type	
1	1	0	/31024/0/1	Boolean	<input type="checkbox"/> <input type="checkbox"/>
2	1	1	/31024/0/2	Boolean	<input type="checkbox"/> <input type="checkbox"/>

Note: If connect LwM2M Bootstrap Server, please enable "Requested Bootstrap", set port to 6683(CoAP) or port 6684(CoAPS).

2. Enable "Modbus To LwM2M Mapping Channel", Modbus Master will select the value send to the remote LwM2M server from Modbus Slave.

Channel Settings

Modbus To LwM2M Mapping Channel

Index: 1

Slave ID: 1

Register Address: 0

Data Type: Boolean

Object ID: /31024/0/1

Object Resource Type: Boolean

PSK Identity: 000000

PSK Key: 123456

Requested Bootstrap:

Lifetime: 300

CoAP Block Size: 512

Enable Verbose Log:

Modbus To LwM2M Mapping Channel

Index	Slave ID	Register Address	Object ID	Object Resource Type
1	1	0	/31024/0/1	Boolean
2	1	1	/31024/0/2	Boolean

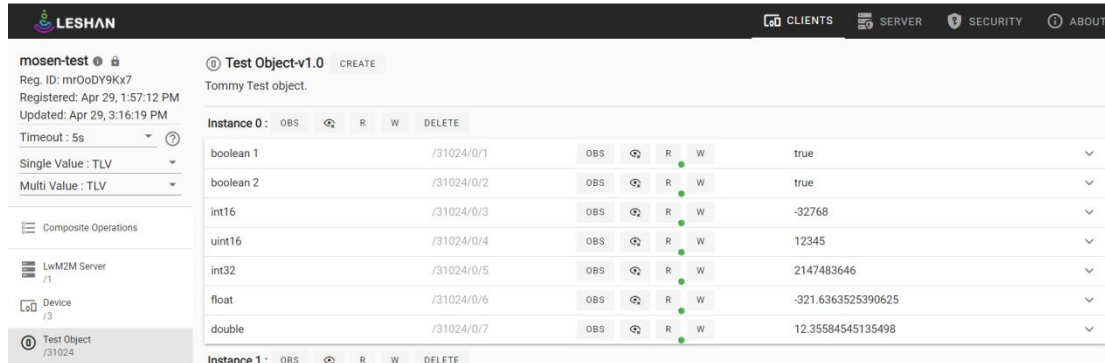
3. Click Save>Save>Apply. (Note: This is a secondary list, it needs to double click save)

4. Go to **Application>Modbus LwM2M>Status**, NR500(Modbus Master) had connected to the remote LwM2M server successfully via LwM2M protocol.

Status		Modbus Transport		X.509 Certificate	
Connection Status					
Index	Enable	Description	Protocol	Status	Uptime
1	true	TCP Setting	TCP Client	Connected	00:02:35

4. Testing

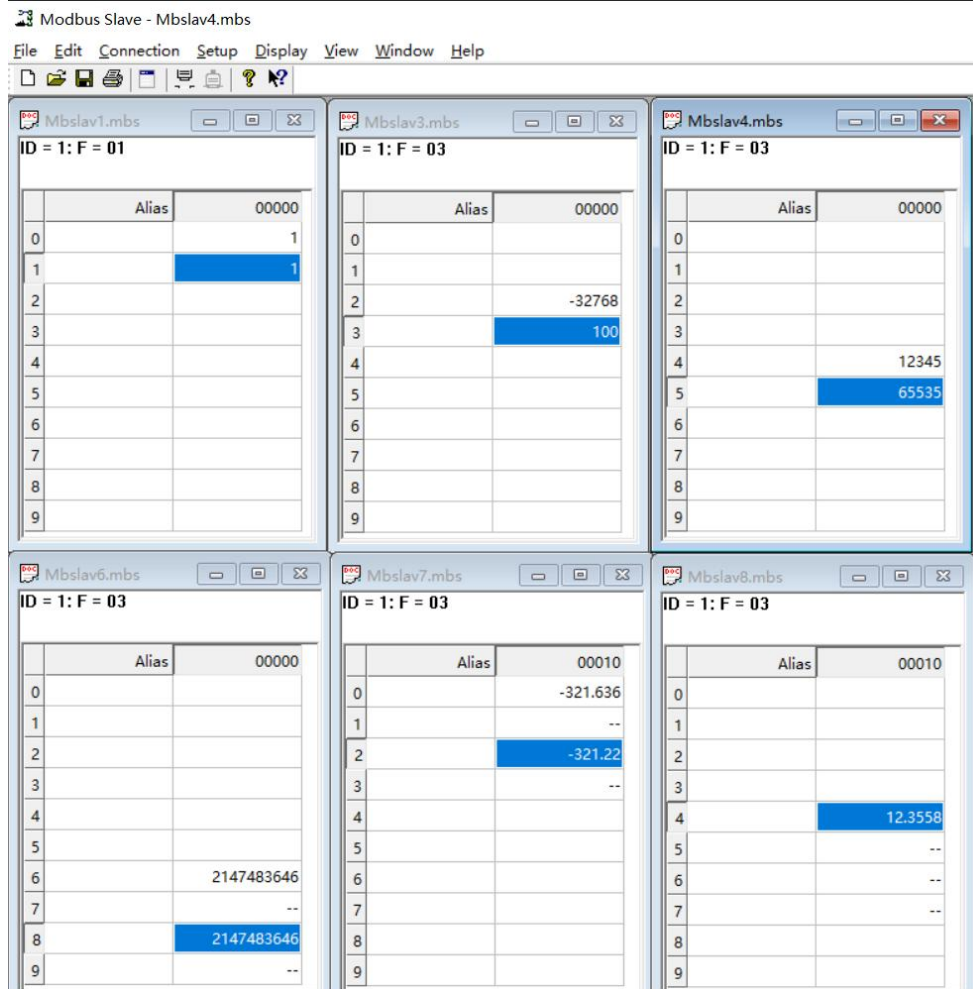
1. Remote LwM2M Server received the data successfully.



The screenshot shows the LESHAN LwM2M Server interface. On the left, there is a sidebar with navigation options: Composite Operations, LwM2M Server, Device, and Test Object. The main area displays the configuration for a Test Object named 'Test Object-v1.0'. It lists various data points with their paths, data types, and current values.

Path	Data Type	Value
/31024/0/1	boolean 1	true
/31024/0/2	boolean 2	true
/31024/0/3	int16	-32768
/31024/0/4	uint16	12345
/31024/0/5	int32	2147483646
/31024/0/6	float	-321.6363525390625
/31024/0/7	double	12.35584545135498

2. Click the "R" to read the data, Click the "W" to write data to Modbus slave. When 'OBS' is enable, value will be update in the real time.



The screenshot shows the Modbus Slave - Mbslav4.mbs interface with six data tables for different slaves. Each table has an 'Alias' column and a 'Value' column. The values are updated in real-time as indicated by the blue highlights.

Slave	Alias	Value
Mbslav1.mbs	00000	1
Mbslav3.mbs	00000	100
Mbslav4.mbs	00000	65535
Mbslav6.mbs	00010	2147483646
Mbslav7.mbs	00010	-321.22
Mbslav8.mbs	00010	12.3558

Test successfully.