

ROUTER TITAN

Nota de aplicación 80

Implementando una pasarela Modbus RTU a SNMP

Implementando una pasarela Modbus RTU a SNMP

Detalles del escenario

Los router Titan disponen de todas las funcionalidades típicas de un router 4G/3G/2G pero además cuentan con una serie de prestaciones adicionales que lo convierten en uno de los routers con más prestaciones del mercado.

Una de las prestaciones adicionales es la capacidad de implementar una pasarela Modbus RTU a SNMP, por ejemplo, para dotar de comunicaciones SNMP a un dispositivo con comunicaciones MODBUS RTU que no disponga de SNMP.

Como siempre, se ilustrará esta capacidad con un ejemplo sencillo.

1. Descripción del ejemplo

En este ejemplo se va a configurar un dispositivo WEBDYN-EASY-TUNNEL para actuar como una pasarela Modbus RTU a SNMP v2C para un PLC que únicamente cuenta con comunicaciones Modbus RTU. El PLC va a estar integrado en una infraestructura donde un gestor SNMP debe poder leer periódicamente las distintas variables del PLC así como poder escribir en alguna de ellas, todo vía SNMP. Además, se debe poder enviar TRAPS SNMP en ciertas ocasiones al gestor SNMP. El PLC dispone de un mapa de 100 registros modbus de tipo WORD, mapeados desde la dirección de registro 1 a la 100.



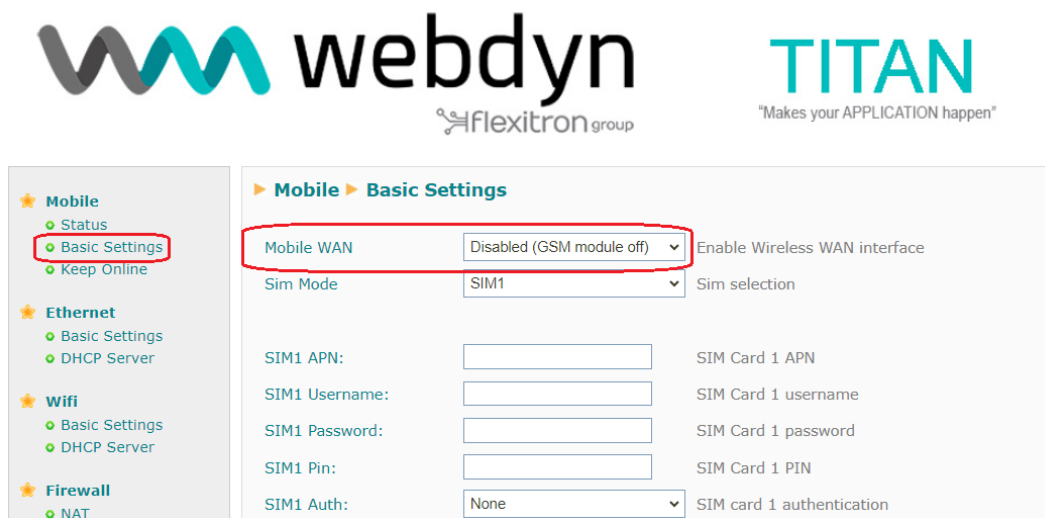
Se pretende el siguiente funcionamiento: las variables modbus del PLC de la dirección modbus 1 a 10 deben poder ser accesibles mediante comando GET SNMP desde el gestor SNMP. También debe poderse escribir un valor desde el gestor SNMP mediante un comando SET SNMP en dichas variables del PLC. Se precisa que el sistema pueda enviar TRAPS SNMP ante ciertas variables. En este caso, a diferencia del caso de modbus TCP donde es el propio PLC quien ordena el envío del TRAP mediante un comando AT enviado por Modbus TCP, el envío del TRAP debe programarse en el SCRIPT dentro del router Titan. Esto es debido a que el router Titan tiene que estar configurado como Modbus RTU MASTER para leer continuamente los registros Modbus del PLC, por lo que no puede configurarse de forma simultánea como modbus RTU SLAVE para recibir comandos desde el PC (como si puede hacerse usando modbus TCP). Para este ejemplo se utilizará SNMP v2c.

2. Configuración del WEBDYN-EASY-TUNNEL

A continuación se describirá en detalle cómo configurar cada sección del WEBDYN-EASY-TUNNEL para llevar a cabo dicho escenario.

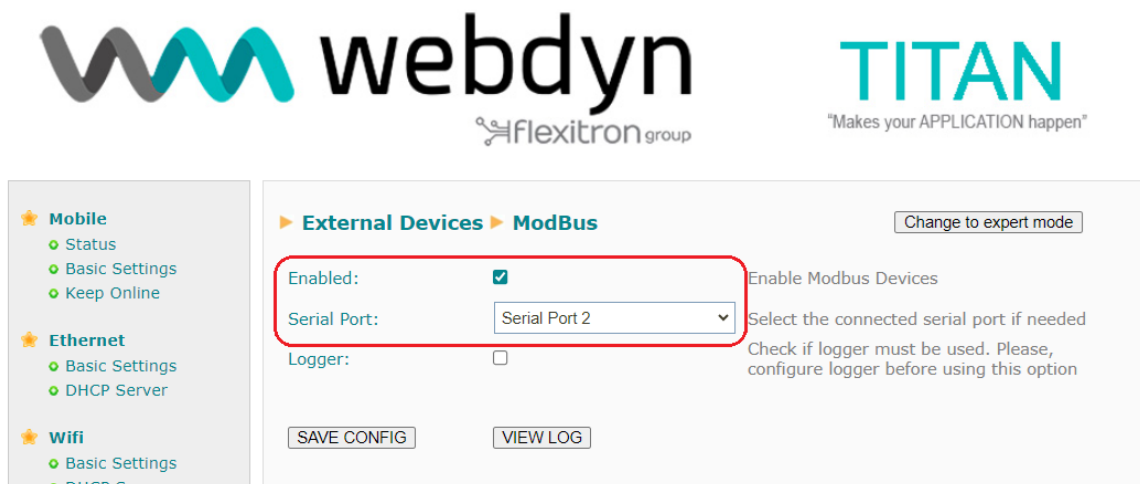
2.1 Configuración Mobile

Para este ejemplo no se precisa uso de tarjeta SIM, por lo que no se va a utilizar. Como el equipo no va a utilizar tarjeta SIM debe configurarse apropiadamente el dispositivo. Para ello, desde el menú de configuración “Mobile - Basic Settings”, en la casilla “Mobile WAN” especificaremos el valor “Disabled (GSM module off)”.



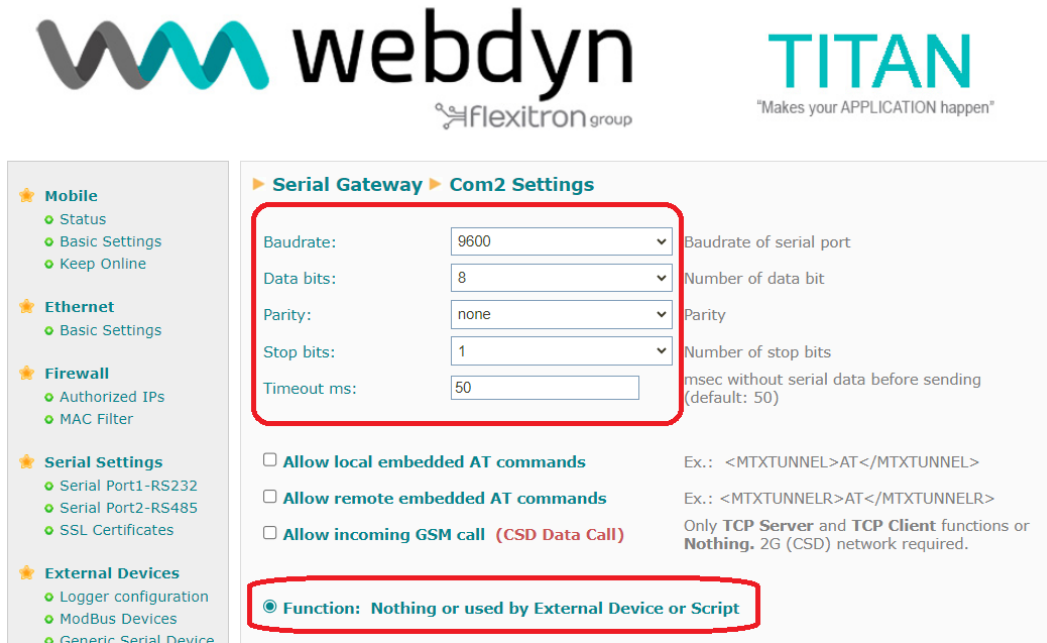
2.2 Configuración del Servicio MODBUS

Para esta aplicación debe activarse el servicio MODBUS, pues éste será utilizado por el dispositivo WEBDYN-EASY-TUNNEL para leer periódicamente los registros modbus del PLC. Vamos a utilizar Modbus RTU por lo que debe activarse el servicio asociando el servicio al puerto serie RS485 (que corresponde con el “Serial Port 2”). Esta configuración se realiza desde el menú “External Devices - Modbus Devices”.



2.3 Configuración del puerto serie RS485

Como el servicio de Modbus está asociado al puerto serie RS485 (Serial Port 2) habrá que configurarlo apropiadamente para la correcta comunicación con el PLC. Si el PLC tiene una configuración de su puerto serie RS485 de 9600,8,N,1, la misma configuración habrá que aplicarla sobre el router Titan:



The screenshot shows the Titan router configuration interface. The top left features the 'webdyn flexitron group' logo, and the top right features the 'TITAN' logo with the tagline 'Makes your APPLICATION happen'. The left sidebar contains a navigation menu with categories: Mobile (Status, Basic Settings, Keep Online), Ethernet (Basic Settings), Firewall (Authorized IPs, MAC Filter), Serial Settings (Serial Port1-RS232, Serial Port2-RS485, SSL Certificates), and External Devices (Logger configuration, ModBus Devices, Generic Serial Device). The main content area is titled 'Serial Gateway > Com2 Settings'. A red box highlights the 'Baudrate' (9600), 'Data bits' (8), 'Parity' (none), 'Stop bits' (1), and 'Timeout ms' (50) fields. Below these fields are three checkboxes: 'Allow local embedded AT commands', 'Allow remote embedded AT commands', and 'Allow incoming GSM call (CSD Data Call)'. A second red box highlights the 'Function: Nothing or used by External Device or Script' radio button.

2.4 Configuración del Servicio MODBUS SLAVE

Las variables modbus que van a ser leídas del PLC van a ser almacenadas en la zona de memoria de usuario MODBUS del WEBDYN-EASY-TUNNEL (zona de memoria que se encuentra desde la dirección de registro 10000 a la 19999). Por ello es por lo que hay que activar el servicio MODBUS TCP SLAVE, para activar esa zona de memoria. Esto se realiza desde el menú "Other - Modbus Slave".

Nota importante: debe activarse el servicio Modbus TCP slave aunque no se utilice. No debe activarse el servicio Modbus RTU Slave, ya que el puerto serie RS485 será utilizado como Master para leer los registros del PLC



The screenshot shows the Titan router configuration interface. The top left features the 'webdyn flexitron group' logo, and the top right features the 'TITAN' logo with the tagline 'Makes your APPLICATION happen'. The left sidebar contains a navigation menu with categories: Mobile (Status, Basic Settings, Keep Online) and Ethernet. The main content area is titled 'Other > ModBus TCP Slave'. A red box highlights the 'Enabled' checkbox (checked) and the 'ModBus TCP Port' field (502).

2.5 Configuración del Servicio SNMP

Es necesario también activar el servicio SNMP v2c. Para ello debe acudir al menú de configuración “Other - SNMP” y configurar la parte apropiada para v2c.



The screenshot shows the 'Other > SNMP' configuration page. A red box highlights the 'Enabled' checkbox (checked), 'SNMP Version' dropdown (set to 'SNMPv2c'), 'UDP Port' (161), and 'Custom OID' (45711.1.1). Other visible fields include 'Community' (public), 'Username', and 'Auth Password'. Descriptive text on the right explains the purpose of each field.

2.6 Configuración del Script de lectura Modbus

El siguiente paso es programar el script de lectura modbus. Este script se encargará básicamente de leer continuamente los registros modbus del PLC e ir almacenándolos en la zona de memoria de usuario del WEBDYN-EASY-TUNNEL. También se utilizará para configurar el modo en el que el WEBDYN-EASY-TUNNEL debe comportarse ante el comando SNMP SET recibidos sobre los registros modbus.



The screenshot shows the 'Other > Titan Scripts v2' configuration page. A dropdown menu is set to 'Current Titan Script' and a 'Load Example' button is visible. The main area contains a code editor with a JavaScript script for reading Modbus registers and sending traps.

```
1 var address="1"; //PLC modbus address
2 var initialRegisterAddress=1; //Initial register address on PLC
3 var trapSent=false;
4
5 //CONFIGURATION OF SNMP SET COMMAND OVER MODBUS OIDS
6 //DEVICE IS CONFIGURED TO WRITE TO THE CORRESPONDING RTU MODBUS REGISTER ON PLC
7 mtx.snmpModbusSetMode(1,address,initialRegisterAddress);
8
9 while (true)
10 {
11 //READ 100 MODBUS REGISTERS FROM PLC, FROM ADDRESS 1 BY USING MODBUS COMMAND 3
12 var res=mtx.modbusRTUGetWords(address,3,initialRegisterAddress,100);
13 //IF READ WAS OK ...
14 if (res!=null)
15 {
16 //WRITE MODBUS REGISTERS INTO USER MEMORY SPACE (STARTS AT ADDRESS 10000)
17 for (var i=0;i<res.length;i++)
18 mtx.modbusSaveWord(i+10000,res[i]);
19 //SEND A SNMP TRAP WHEN FIRST MODBUS REGISTER == 10
20 if (res[0]==10)
21 {
22 if (trapSent==false)
23 {
24 //PARAMETERS: OID,message,severity,ip,port,community
25 mtx.trapSend("1.2.3.4.5.6","alert",1,"192.168.1.28",162,"public");
26 mtx.println("TRAP SENT");
27 trapSent=true;
28 }
29 }
30 else
31 trapSent=false;
32 }
33 else
34 mtx.println("ERROR READING PLC");
35
36 mtx.pause(500);
37 }
```

A continuación el script explicado paso a paso.

```
1 var address="1";           //PLC modbus address
2 var initialRegisterAddress=1; //Initial register address on PLC
3 var trapSent=false;
4
```

En estas líneas se están definiendo 3 variables. La variable "address" especifica la dirección modbus RTU del PLC ("1"). La segunda variable "initialRegisterAddress" indica la dirección del registro inicial a leer del PLC, que será la 1. La tercera es una variable auxiliar utilizada más adelante.

```
5 //CONFIGURATION OF SNMP SET COMMAND OVER MODBUS OIDs
6 //DEVICE IS CONFIGURED TO WRITE TO THE CORRESPONDING RTU MODBUS REGISTER ON PLC
7 mtx.snmpModbusSetMode(1,address,initialRegisterAddress);
```

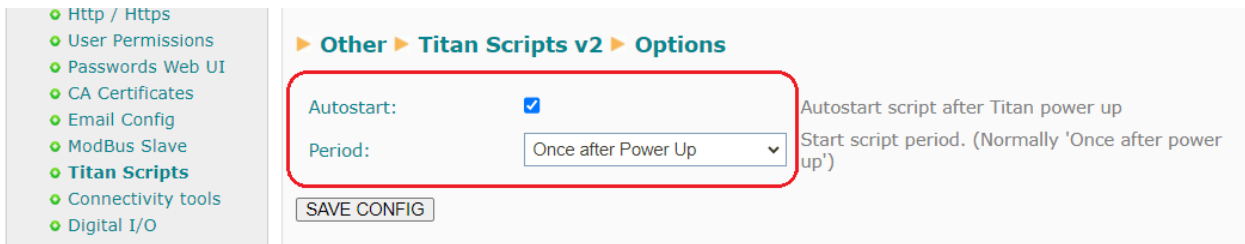
En la llamada a la función snmpMosbusSetMode lo que se hace es configurar el modo de funcionamiento que hará el dispositivo WEBDYN-EASY-TUNNEL al recibir un comando SNMP SET sobre un OID de la memoria de usuario modbus. Un valor "1" hace que el WEBDYN-EASY-TUNNEL, antes de almacenar el valor recibido por SNMP es su memoria interna, ejecute un comando modbus de escritura para escribirlo en el dispositivo modbus con dirección RTU "1" (en este caso el PLC) y desde el registro "initialRegisterAddress".

```
9 while (true)
10 {
11     //READ 100 MODBUS REGISTERS FROM PLC, FROM ADDRESS 1 BY USING MODBUS COMMAND 3
12     var res=mtx.modbusRTUGetWords(address,3,initialRegisterAddress,100);
13     //IF READ WAS OK ...
14     if (res!==null)
15     {
16         //WRITE MODBUS REGISTERS INTO USER MEMORY SPACE (STARTS AT ADDRESS 10000)
17         for (var i=0;i<res.length;i++)
18             mtx.modbusSaveWord(i+10000,res[i]);
19         //SEND A SNMP TRAP WHEN FIRST MODBUS REGISTER == 10
20         if (res[0]==10)
21         {
22             if (trapSent==false)
23             {
24                 //PARAMETERS: OID,message,severity,ip,port,community
25                 mtx.trapSend("1.2.3.4.5.6","alert",1,"192.168.1.28",162,"public");
26                 mtx.println("TRAP SENT");
27                 trapSent=true;
28             }
29         }
30         else
31             trapSent=false;
32     }
33     else
34         mtx.println("ERROR READING PLC");
35
36     mtx.pause(500);
37 }
```

Por último está este bucle while infinito. En el continuamente se leen 100 registros modbus del PLC con el comando modbus 3. Si el resultado de la lectura de los 100 registros es correcto, se almacenan los registros leídos en la memoria interna de usuario modbus, cuya dirección inicial es 10000.

Adicionalmente, se comprueba el valor del primer registro modbus leído. En el caso de que éste tenga un valor “10” se envía un TRAP SNMP.

Por último sólo restaría configurar la autoejecución del script, esto es, configurar el WEBDYN-EASY-TUNNEL para que inicie automáticamente el script cada vez que se reinicie. Para ello:



Una vez reiniciado el dispositivo debería funcionar el escenario de pasarela Modbus TCP- SNMP

2.7 Probando el escenario.

Es sencillo probar el correcto funcionamiento de la pasarela utilizando, por ejemplo, el software “ManageEngine MibBrowser” junto con un emulador de esclavo modbus como “Modbus Slave”. Es posible comprobar el funcionamiento viendo como, variando los valores de los registros en el dispositivo PLC emulado con “Modbus Slave”, estos pueden ser leídos mediante comandos SNMP GET realizados desde el software ManageEngine MibBrowser.

