

ROUTER TITAN Nota de aplicación 69

Sending data to eSight platform

www.webdyn.com

Sending data to eSight platform

1. 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 datalogger donde el router Titan almacena en su memoria no volátil distintos tipos de registros en un formato JSON. Estos registros almacenados pueden provenir de lecturas MODBUS, o de capturas de datos SERIE por sus puertos RS232 / RS485, de posiciones GPS, etc Estos registros de tipo JSON son almacenados por el router Titan en su memoria interna no volátil para ser posteriormente enviados a una plataforma remota mediante comunicaciones tipo HTTP, HTTPS, MQTT, MQTTS, FTP, FTPS.

Como se indica, el router Titan almacena en su memoria interna dichos registros JSON en un formato predeterminado y propietario. En ocasiones, esto puede ser un problema para la comunicación con aquellas plataformas que esperan recibir la información en un determinado formato (es decir, con un formato distinto de JSON al utilizado por defecto por el router Titan).

En esta nota de aplicación se desarrollará el ejemplo completo para el envio de datos a la conocida plataforma de MRI eSight (https://www.mrisoftware.com/products/esight-energy-management-software).

En este caso, vamos a suponer que se deben leer 2 registros de 2 PLCs con comunicaciones Modbus conectados a un Webdyn-EasyTunnel por su puerto RS485.



En concreto se pretende lo siguiente: cada minuto, el dispositivo Webdyn EasyTunnel debe leer los registros Modbus con dirección 40000 y 40001 del PLC1 y los registros 40000 y 40001 del PLC2. En ambos PLCs el registro 40000 corresponde con la temperatura medida y el registro 40001 con la humedad. Las lecturas realizadas deben almacenarse en la memoria interna no volátil del Webdyn EasyTunnel (en su datalogger) y éste debe enviar dichos datos leídos a la plataforma MRI eSight siempre que sea posible (haya cobertura, conectividad IP, ...) La comunicación con los PLCs es a través de un bus RS485 con una configuración 9600,8,N,1.

2. Configuración de la Mobile WAN

El Webdyn EasyTunnel debe comunicarse con la plataforma MRI eSight a través de comunicaciones 4G/3G/2G, por lo que la sección "Mobile>Basic Settings" debe configurarse apropiadamente en función de la tarjeta SIM utilizada

Basic Settings	Mobile WAN	Enabled (IP active)	~	Enable Wireless WAN interface
Keep Unline	Sim Mode	SIM1	~	Sim selection
thernet Basic Settings				
irowall	SIM1 APN:	movistar.es		SIM Card 1 APN
Authorized IPs	SIM1 Username:	MOVISTAR		SIM Card 1 username
Serial Settings	SIM1 Password:	•••••		SIM Card 1 password
Serial Port1-RS232	SIM1 Pin:			SIM Card 1 PIN
SSL Certificates	SIM1 Auth:	None	~	SIM card 1 authentication
External Devices				
 Logger configuration ModBus Devices 	SIM2 APN:			SIM Card 2 APN
Generic Serial Device	SIM2 Username:			SIM Card 2 username
• IEC102 Meter	SIM2 Password:			SIM Card 2 password
GPS Receiver	SIM2 Pin:		-	SIM Card 2 PIN
Plugins	SIM2 Auth:	Auto	~	SIM card 2 authentication
Generic				
Other		1. · · · · · · · · · · · · · · · · · · ·		
Al Command DynDns	Network selection:	Auto (4G/3G/2G)	~	Network selection
• Private DynDns				
Sms control	DNS selection:	Get DNS from Operator	~	
Time Servers	DNS1:	8.8.8.8		Preferred DNS1
Remote Console Some	01100	0044	-	Desferred DNCO
• • • • • • • • • • • • • • • • • • •	DINS2:	0.0.4.4		Prefetted DNDZ

3. Configuración del puerto serie RS485

La comunicación con los dos PLCs se realizará a través de puerto serie RS485 a 9600,8,N,1, por lo que debe configurarse la sección "Serial Settings > Serial Port2-RS485" ajustando los parámetros como se indica a continuación.

* Mobile	Serial Gateway	Com2 Settings					
Basic Settings	Baudrate:	9600 ~	Baudrate of serial port				
 Keep Online 	Data bits:	8	Number of data bit				
🔶 Ethernet	Parity:	none	Parity				
• Basic Settings	Stop bits:	1 ~	Number of stop bits				
 Firewall Authorized IPs 	Timeout ms:	50	msec without serial data before sending (default: 50)				
 Serial Settings Serial Port1-RS232 Serial Port2-RS485 SSL Certificates 	Allow local embed Allow remote emb	ded AT commands edded AT commands	Ex.: <mtxtunnel>AT</mtxtunnel> Ex.: <mtxtunnelr>AT</mtxtunnelr>				
 External Devices Logger configuration ModBus Devices Generic Serial Device Temperature Sensor IEC102 Meter GPS Receiver 	Allow incoming GS Function: Nothing	M call (CSD Data Call) or used by External Device	Only TCP Server and TCP Client functions or Nothing. 2G (CSD) network required.				
• or or negative	O Function: Serial -	IP Gateway (TCP Server)					
Plugins • Generic	TCP Local Port:	20011	Listening TCP Port (1 65535)				
Other AT Command Durphase	Temporal client RS232 Temporal client Wakeup		Check if you need a temporal TCP Client whe data is present at serial port. DDHHMM. Example: XX2200 starts a tempor client every day at 22:00				
Private DynDns	Temporal client time:	60	Seconds for temporal client				
 Sms control Periodic Autoreset 	Temporal client Random	0	Seconds. Random time for temporal client Wakeup				
• Time Servers • Remote Console	SSL/TLS enabled		SSL/TLS Enabled (SSL Certs needed)				

4. Configuración del Logger

El siguiente paso es configurar el datalogger interno del Webdyn EasyTunnel. Para ello debe acudirse al menú "External Devices > Logger configuration". La configuración debería ser similar a la que se muestra en la siguiente figura:

🔶 Mobile	▶ External Devices ▶ Logger						
Status Basic Settings Keen Opling	ID:	ID000	Optional. Device identification				
V Keep Online	Send mode:	FIFO 🗸	Send mode (normally FIFO)				
Ethernet	Time format:	std (dd/mm/yyyy HH:mm:ss) 🗸	Time format used in timestamp logger data				
• DHCP Server	Use script:		Check for customized json using 'Json Transformer Script' in Script section.				
🔶 Wifi	Use array:		Check if you want to send more than one ISON per transmition				
Basic SettingsDHCP Server	Check date:	Save data in Logger only if date has been set (check Time Servers)					
🔶 Firewall o NAT	Communication mod	e: WEB PLATFORM (HTTP R	REST)				
 Firewall NAT Authorized IPs MAC Filter 	Communication mod Enabled:	e: WEB PLATFORM (HTTP F	REST) Communication mode HTTP enabled				
 Firewall NAT Authorized IPs MAC Filter Routes 	Communication mod Enabled: Mode:	e: WEB PLATFORM (HTTP F	REST) Communication mode HTTP enabled Method of sending data				
 Firewall NAT Authorized IPs MAC Filter Routes Serial Settings 	Communication mod Enabled: Mode: Custom header1:	e: WEB PLATFORM (HTTP F	EST) Communication mode HTTP enabled Method of sending data Optional. Custom header1. For example: Content-type:application/ison				
 Firewall NAT Authorized IPs MAC Filter Routes Serial Settings Serial Port1-RS232 Serial Port2-RE485 	Communication mod Enabled: Mode: Custom header1: Custom header2:	e: WEB PLATFORM (HTTP R HTTPS POST (JSON)	EST) Communication mode HTTP enabled Method of sending data Optional. Custom header1. For example: Content-type;application/json Optional. Custom header2. For example: IDENTITY_KEY;YOUR_KEY				
 Firewall NAT Authorized IPs MAC Filter Routes Serial Settings Serial Port1-RS232 Serial Port2-RS485 SSL Certificates 	Communication mod Enabled: Mode: Custom header1: Custom header2: Custom header3:	e: WEB PLATFORM (HTTP R	Communication mode HTTP enabled Method of sending data Optional. Custom header1. For example: Content-type;application/json Optional. Custom header2. For example: IDENTITY_KEY;YOUR_KEY Optional. Custom header3.				
 Firewall NAT Authorized IPs MAC Filter Routes Serial Settings Serial Port1-RS232 Serial Port2-RS485 SSL Certificates External Devices 	Communication mod Enabled: Mode: Custom header1: Custom header2: Custom header3: Server:	e: WEB PLATFORM (HTTP R HTTPS POST (JSON) ~	Communication mode HTTP enabled Method of sending data Optional. Custom header1. For example: Content-type;application/json Optional. Custom header2. For example: IDENTITY_KEY;YOUR_KEY Optional. Custom header3. Destination URL. Example: www.mydomain.com/setdata.php				
 Firewall NAT Authorized IPs MAC Filter Routes Serial Settings Serial Port1-RS232 Serial Port2-RS485 SSL Certificates External Devices Logger configuration 	Communication mod Enabled: Mode: Custom header1: Custom header2: Custom header3: Server: Server Username:	e: WEB PLATFORM (HTTP R	Communication mode HTTP enabled Method of sending data Optional. Custom header1. For example: Content-type;application/json Optional. Custom header2. For example: IDENTITY_KEY;YOUR_KEY Optional. Custom header3. Destination URL. Example: www.mydomain.com/setdata.php Optional. Blank if no server authentication required				

• Temperature Sensor

Cabe destacar los siguientes parámetros:

• "ID" proporciona al registrador una identificación única. Sirve como la primera parte de un identificador único (código de importación eSight)

- "Send mode" seleccionar FIFO.
- "Time format" es necesario seleccionar formato "std".
- "Check date" activamos la casilla.
- "Mode" seleccionamos HTTPS POST (JSON).
- "Server" aquí introduciremos la URL del servidor, ingrese su webhook eSight en el cuadro.

5. Configuración de la sección Modbus

En esta sección de configuración "External Devices > Modbus Devices" configuraremos las lecturas Modbus que deben realizarse sobre los 2 PLCs

 Mobile Status Basic Settinge 	External De	vices 🕨	ModBus	RTU / TO	СР						
• Keep Online	Enabled:	5	2			Enable Mod	lbus Devices]
 Ethernet Basic Settings DHCP Server Wifi Basic Settings DHCP Server 	Serial Port: Logger: SAVE CONFIG		Serial Port 2		*	Select the of Check if log Please, con option	connected se gger must be figure logger	rial port used before	t if nee	eded this	
 Firewall NAT 	Dev. name	Addr.	Comm.	Start @	Nun	a word/bit	Reg Type	Period			
Authorized IPs MAC Filter	PLC1	1	0x04	40000	TTU	2	WORD	1	Edit	Del	Test
• Routes	PLC2	2	0x04	40000		2	WORD	1	Edit	Del	Test
 Serial Settings Serial Port1-RS232 Serial Port2-RS485 SSL Certificates External Devices Logger configuration ModBus Devices Generic Serial Device Temperature Sensor IEC102 Meter EV Charge Point 	Device name / II Address: Command: Start: Number Words / Reg Type: Period:	D: [[Bits: [[PLC2 2 0x04 40000 2 WORD 1		 	Insert the o Modbus RT Modbus rea Address of Words for o 0x01/0x02 Type of reg Read period	device name U address or ad command the first regi command 0xi isters for cor d (minutes)	or ID IP:port ster 03/0x04 mmand	addre . Bits 0x03/1	ess for 0x04	
IPSecOpenVPN ClientOpenVPN Server	SAVE DEVICE (Max 40 n	nodbus dev	ices)							

Debe habilitarse el servicio Modbus activando la casilla "Enabled". Debe seleccionar "Serial Port 2", pues la lectura se realizará a través del puerto RS485. Y también debe activarse la casilla "Logger",

pues los registros Modbus leídos de los PLCs deben almacenarse en el datalogger interno del Webdyn EasyTunnel.

También deben crearse dos dispositivos que en este ejemplo hemos llamado PLC1 y PLC2, servirán como segunda parte de un identificador único (código de importación eSight). En "address" debe indicarse la dirección modbus de cada uno de los PLCs, por lo que en PLC1 pondremos la dirección 1 y en PLC2 la dirección 2. El comando modbus que utilizaremos para la lectura de los registros 40000 y 40001 será el 0x04, por lo que seleccionaremos 0x04 en el campo "Command". El registro iniciar a leer en ambos PLCs es el registro 40000, pues indicaremos en el campo "Start" el valor 40000. Pretendemos leer 2 registros modbus de cada PLC (el 40000 y 40001) por lo que en el campo "Number Words" indicaremos 2. El tipo de registro "Reg Type" lo seleccionaremos como "WORD" y en "Period" seleccionaremos 1, pues queremos lecturas de los registros Modbus cada minuto.

6. Configuración en eSight

El formato de JSON estándar generado por el Webdyn EasyTunnel con los datos leídos tendrá un formato como el que sigue:

{"IMEI":"869101054286683","TYPE":"MODB","TS":"2022-11-11T12:17:00Z","ID":"PLC1","A":"1","ST": "40000","N":"2","V":[225,62],"P":"ID000"},

{"IMEI":"869101054286683","TYPE":"MODB","TS":"2022-11-11T12:17:01Z","ID":"PLC2","A":"2","ST": "40000","N":"2","V":[225,62],"P":"ID000"}

Será necesario desarrollar un pequeño script para adaptarlo a las necesidades de la plataforma eSight. A continuación, se muestra un código que puede servir como ejemplo:

let inJSON = JSON.parse(inData);

let meter_id = inJSON.ID; let site_name = inJSON.P; let timestamp = inJSON.TS; let value = inJSON.V[0];

// May need to change/remove indexing

let r = new eS.MeterReading(); r.MeterIdentifier = site_name + "_" + meter_id; r.Timestamp = timestamp; r.Value = value;

outData.push(r)
// Un-comment above line when device is setup

¿Más dudas?

Escríbanos sus consultas a soporte@matrix.es