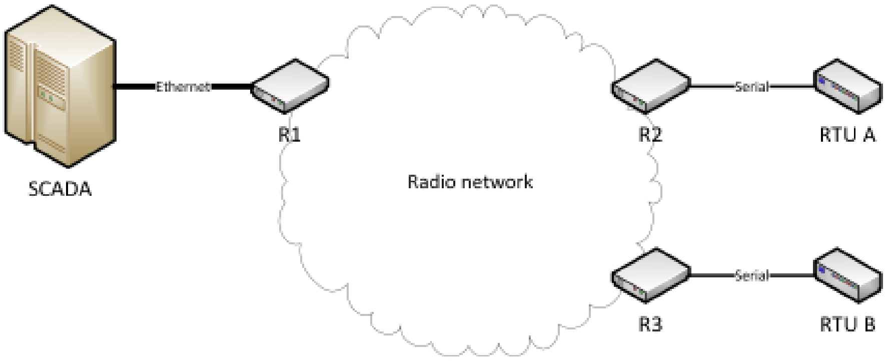
SATELLAR Protocols

Example 1. : DNP3 with TCP, UDP and serial port

In this example, SATELLARs are used to enable DNP3 communication between a Supervisory Control And Data Acquisition (SCADA) device and two Remote Terminal Units (RTU). The SCADA is connected to SATELLAR R1 with an Ethernet cable, and the two RTUs to their SATELLARs with a serial cable. SATELLAR R1 has radio connectivity to both R2 and R3, and there can be repeaters between. The setup can be seen in figure down below.



The devices will have the following addresses:

Device	IP Address	RMAC Address	DNP3 Address
SCADA	192.168.1.100		255
R1	192.168.1.1	1	
R2	192.168.2.1	2	
R3	192.168.3.1	3	
RTU A			2
RTU B			3

In this topology, R1 is the master device and the other two are slave devices. R1 will have a TCP server that listens to the DNP3 messages from the SCADA sent over TCP. It will relay the messages as UDP packets to R2 and R3. R2 and R3 will then relay the messages through the serial port to the connected RTU. If RTU A sends a message to the SCADA, it is written to the serial port of R2. R2 will send it as a UDP packet to R1, which will in turn write the message to the SCADA through the open TCP connection.

The Application Routing parameters for the three SATELLARs can be seen in the table below. The serial port parameters are not included; they will just be set so that they are the same as in the RTUs.
The Application Listening Port of R1 is 20000, so the SCADA needs to open a TCP connection to 192.168.1.1:20000. All SATELLARs must have the same transport protocol, in this case UDP. The master sends to destination port 2006, so both the slaves must have listening port set to 2006. Correspondingly, both slave devices have destination ports 2005 and R1 has listening port 2005.

Device	**R1**	**R2**	**R3**
Application Protocol	DNP3	DNP3	DNP3
Application Transport Protocol	TCP	Serial Port	Serial Port
Application Listening Port	20000	(No effect)	(No effect)
Serial Port	(No effect)	RS-232	RS-232
Transport Protocol For Substation Data	UDP	UDP	UDP
Destination Port For Substation Data	2006	2005	2005
Listening Port For Substation Data	2005	2006	2006
Address Mapping	Application Address to RMAC	Manual	Manual
Address Mapping Row	(empty)	255 192.168.1.1	255 192.168.1.1

Both slave SATELLARs' RMAC addresses match the DNP3 addresses of the RTUs, so R1 can use Application Address to RMAC Address Mapping. Messages sent to the DNP3 address 2 will be routed first to R2, and then to RTU A. Because the SCADA's address is 255, both slaves need to use manual mapping. They both need just one Address Mapping Row: 255 192.168.1.1

This means that the DNP3 messages to the destination address 255 will be routed to R1, which will in turn relay the message to the SCADA.

Variations to the example

The example uses UDP to send the messages over the radio. This is generally recommended, since UDP uses less radio resources than TCP and is also faster. But if a slower but more secure connection is desired, TCP can also be used to transport the messages over the radio. That can be done by simply changing Transport Protocol For Substation Data to TCP in every SATELLAR. No other settings need to be changed, either in the SATELLARs or the other devices in the network.

Exactly the same example works also, if every SATELLAR has Modbus RTU selected as the Application Protocol. In thiscase the SCADA and RTUs must use Modbus RTU.
In the example above, the RTUs are connected to their respective devices through the serial port. But if they were also connected to the slave devices through Ethernet, then both R2 and R3 would need to change the Application Transport Protocol to TCP and also set a port that the RTUs could use. No other settings need to be changed.

The protocol could be changed to Modbus RTU, IEC101 or Custom Protocol and it would function exactly the same way.

Example 2: Modbus TCP and Modbus TCP/RTU conversion

This example uses the same network topology as the example 1, but this time the SCADA and the RTU A use Modbus TCP and the RTU B uses Modbus RTU. This example shows how to configure the SATELLAR devices so that the SCADA only needs to query R1 with Modbus TCP to get replies from RTU A and B.
The addresses of the different devices look like this:

Device	IP Address	RMAC Address	Modbus Address
SCADA	192.168.1.100		1
R1	192.168.1.1	1	
R2	192.168.2.1	2	
R3	192.168.3.1	3	
RTU A	192.168.2.2		2 (Modbus TCP)
RTU B			3 (Modbus RTU)

R1 will be configured to route Modbus TCP to both substations. Because RTU A uses Modbus TCP, the target IP address is the address of the RTU. RTU B uses Modbus RTU, so the target IP is the address of R3. R3 will have Modbus TCP/RTU conversion enabled. R2 will not need any Application Routing settings. The settings of the other two SATELLAR devices can be seen in table below:

Device	R1	R2
Application Protocol	Modbus TCP	Modbus TCP
Application Transport Protocol	TCP	Modbus RTU (Serial Port)
Application Listening Port	502	(No effect)
Serial Port	(No effect)	RS
Transport Protocol For Substation Data	TCP	TCP
Destination Port For Substation Data	502	(No effect)
Listening Port For Substation Data	(No effect)	502
Address Mapping	Manual	Point-to-point
Address Mapping Table	2 192.168.2.2 3 192.168.3.1	(Empty)

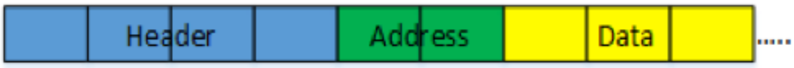
The setup is a bit different compared to the Example 1. Modbus does not use spontaneous transmissions, so the substations send replies only when they are queried. This is why R1 needs only a destination port and R3 only a listening port. R3 does not need any addresses in the mapping table, since it sends all replies over the open TCP connection and it never sends anything spontaneously.

When the SCADA sends a Modbus message to the RTU A, it will first be received by the R1. It analyzes the destination address and opens a TCP connection to the RTU A. The reply will be sent back to the SCADA through the open TCP connection. The R2 only routes the IP packets without any analysis.

When the SCADA sends a Modbus message to the RTU B, the R1 will route the TCP connection to the R3. The R3 will receive the message, convert it to Modbus RTU and write it out of the serial port.

Example 3: Custom Protocol

In this example the protocol that is in use contains a 4-byte header, a 2-byte address and a variable amount of data:



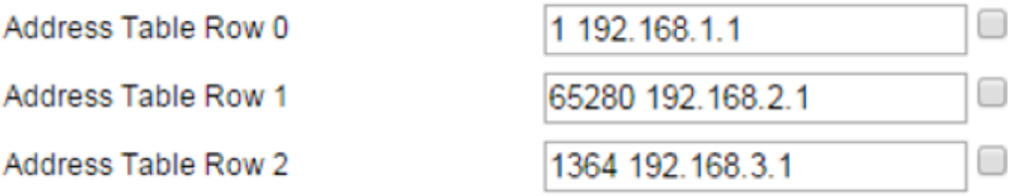
Offset should be set to 4 and Address Length to 16 bits. The following messages would then be interpreted as having the following addresses:

- \01\02\00\00\00\01: Address 1 (\00\01)
- \01\02\07\FF\FF\00\00\00: Address 65280 (\FF\00)
- \01\02\5C\70\05\42\00\00: Address 1364 (\05\42)

In this example the three example messages should be sent to the following IP addresses:

- \01\02\00\00\00\01: 192.168.1.1
- \01\02\07\FF\FF\00\00\00: 192.168.2.1
- \01\02\5C\70\05\42\00\00: 192.168.3.1

Then the mapping table for application routing would look like this:



Address Table Row 0	1 192.168.1.1	☐
Address Table Row 1	65280 192.168.2.1	☐
Address Table Row 2	1364 192.168.3.1	☐