

# Module <SelfSystem> of subsystem “Protocols”

<i>Module:</i>	SelfSystem
<i>Name:</i>	OpenSCADA system own protocol
<i>Type:</i>	Protocol
<i>Source:</i>	prot_SelfSystem.so
<i>Version:</i>	0.9.2
<i>Author:</i>	Roman Savochenko
<i>Translated:</i>	Maxim Lysenko
<i>Description:</i>	OpenSCADA system own protocol, it supports the basic functions.
<i>License:</i>	GPL

## Contents table

<a href="#">Module &lt;SelfSystem&gt; of subsystem “Protocols”</a> .....	1
<a href="#">Introduction</a> .....	1
<a href="#">1. The syntax of the protocol</a> .....	2
<a href="#">2.The internal structure of an outgoing protocol</a> .....	3

## Introduction

The module of the transport protocol SelfSystem is designed to reflect the interface management of OpenSCADA system to the network, to provide an opportunity to the external systems to interact with the OpenSCADA system, as well as for the interaction of the stations constructed on the basis of OpenSCADA among themselves.

The first experience of using the functions of this module was the support of remote configuration of one OpenSCADA station from another through the network, by means of the module of configuration [QTCfg](#).

# 1. The syntax of the protocol

The protocol is built on the mechanism of request-response. Requests and their structure are summarized in Table 1.

**Table 1** Structure of the request.

<p><b>Requests</b></p> <p><b>REQ:</b> «SES_OPEN &lt;user&gt; &lt;password&gt;\n»  <b>REZ OK:</b> «REZ 0 &lt;ses_id&gt;\n»  <b>REZ ERR:</b> «REZ 1 Auth error. User or password error.\n»  The request for the opening of the session on behalf of the user &lt;user&gt; with the password &lt;password&gt;. In case of success it will be received the session identifier, otherwise – the code and the error message.</p>
<p><b>REQ:</b> «SES_CLOSE &lt;ses_id&gt;\n»  <b>REZ:</b> «REZ 0\n»  Closure of the session. The result is always successful.</p>
<p><b>REQ 1:</b> «REQ &lt;ses_id&gt; &lt;req_size&gt; \n &lt;control interface command&gt;”  <b>REQ 2:</b> «REQDIR &lt;user&gt; &lt;password&gt; &lt;req_size&gt; \n &lt;control interface command&gt;”  <b>REZ OK:</b> «REZ 0 &lt;rez_size&gt; \n &lt;control interface command result&gt;”  <b>REZ ERR:</b> «REZ 1 Auth error. Session is not valid.\n»  <b>REZ ERR:</b> “REZ 2 &lt;control interface err&gt;”  The main requests: the session and the direct are implemented by sending the standard command of <a href="#">OpenSCADA control interface</a> to the field &lt;control interface command&gt;. As the result will it be received an answer from the management interface &lt;control interface command result&gt; or one of the errors.</p>
<p><b>REQ:</b> “ERR REQUEST”  <b>REZ ERR:</b> «REZ 3 Command format error.\n»  Any invalid request.</p>

Protocol supports the package of traffic. only the data of the management interface is to be packed <control interface command> and <control interface command result>. The fact of the arrival of packaged request or response is determined by the negative value of the size of the request <req\_size> or response <rez\_size>.

To control the parameters of the package the module provides the configuration form (Fig. 1).

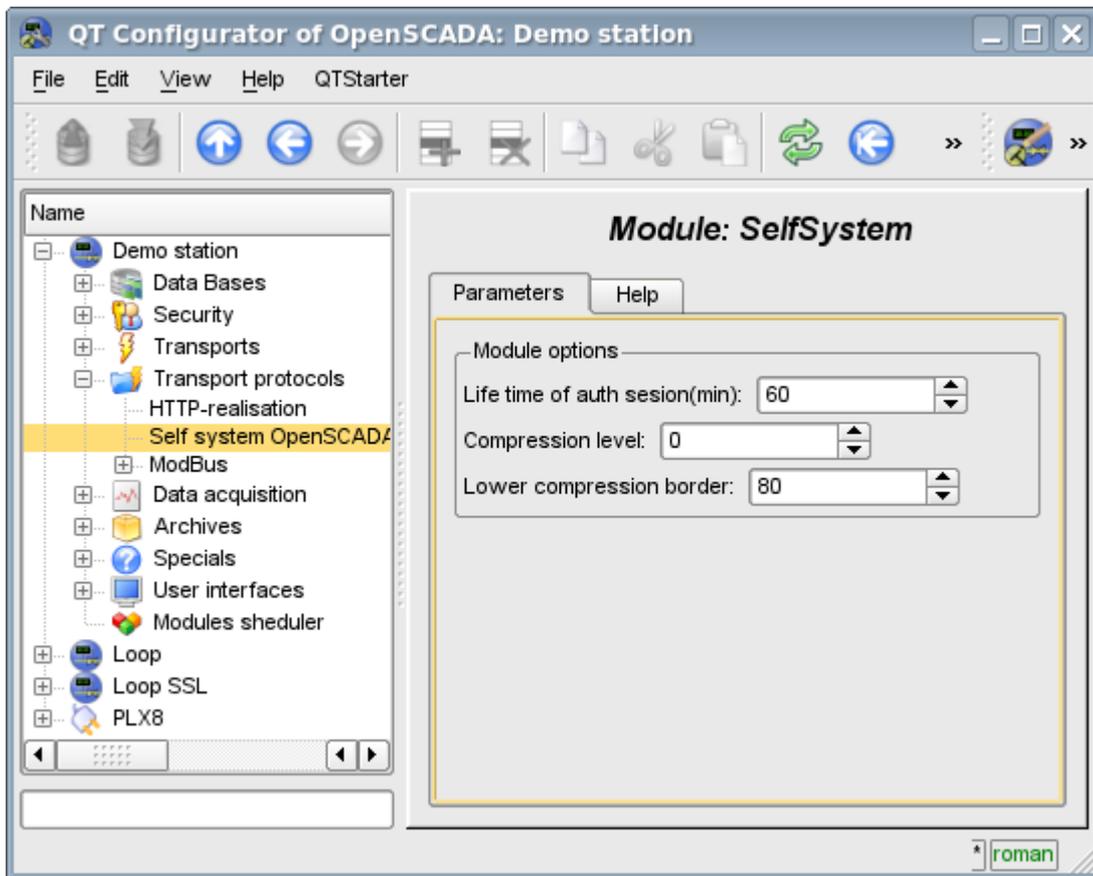


Fig.1. The form of the configuration of the package parameters.

On this form, you can specify:

- the lifetime of the authentication session;
- level of compression of the protocol, ranging from 0 to 9 (0-disable compression-1-optimal in performance and quality compression level );
- lower threshold for the compression using, turns off the compression of small requests.

## 2.The internal structure of an outgoing protocol

The internal structure is formed by means of the tree of XML requests of the language [OpenSCADA control interface](#) with the reservation of the redundant additional service attributes of the protocol in the root tag:

- *rqDir* — sign of the sending the message passing the procedure of the opening the session (0-open session, 1-send immediately);
- *rqUser* — user;
- *rqPass* — password.

The result of the request is the tree of XML language of the management interface of OpenSCADA.