

Модуль подсистемы “Транспорты” <Sockets>

Модуль:	Sockets
Имя:	Сокеты
Тип:	Транспорт
Источник:	tr_Sockets.so
Версия:	1.5.1
Автор:	Роман Савоченко
Описание:	Предоставляет транспорт, основанный на сокетах. Поддерживаются интернет и unix сокеты. Интернет сокет использует TCP и UDP протоколы.
Лицензия:	GPL

Оглавление

Модуль подсистемы “Транспорты” <Sockets>	1
Введение	1
1. Входящие транспорты	2
2. Исходящие транспорты	4

Введение

Модуль транспорта Sockets предоставляет в систему поддержку транспортов, основанных на сокетах. Поддерживаются входящие и исходящие транспорты, основанные на интернет сокетах: TCP, UDP и UNIX сокет. Добавить новые входящие и исходящие сокеты можно посредством конфигурации транспортной подсистемы в любом конфигураторе системы OpenSCADA.

1. Входящие транспорты

Сконфигурированный и запущенный входящий транспорт открывает серверный сокет для ожидания соединения клиентов. В случае с UNIX сокетом создаётся файл UNIX сокета. Сокеты TCP и UNIX являются многопоточными, т.е. при подключении клиента к сокетам данных типов создаётся клиентский сокет и новый поток, в котором производится обслуживание клиента. Серверный сокет в этот момент переходит к ожиданию запросов от нового клиента. Таким образом достигается параллельное обслуживание клиентов.

Каждый входящий сокет обязательно связывается с одним из доступных транспортных протоколов, которому передаются входящие сообщения. В связке с транспортным протоколом поддерживается механизм объединения кусков разрозненных при передаче запросов.

Диалог конфигурации входящего сокета изображён на рис.1.

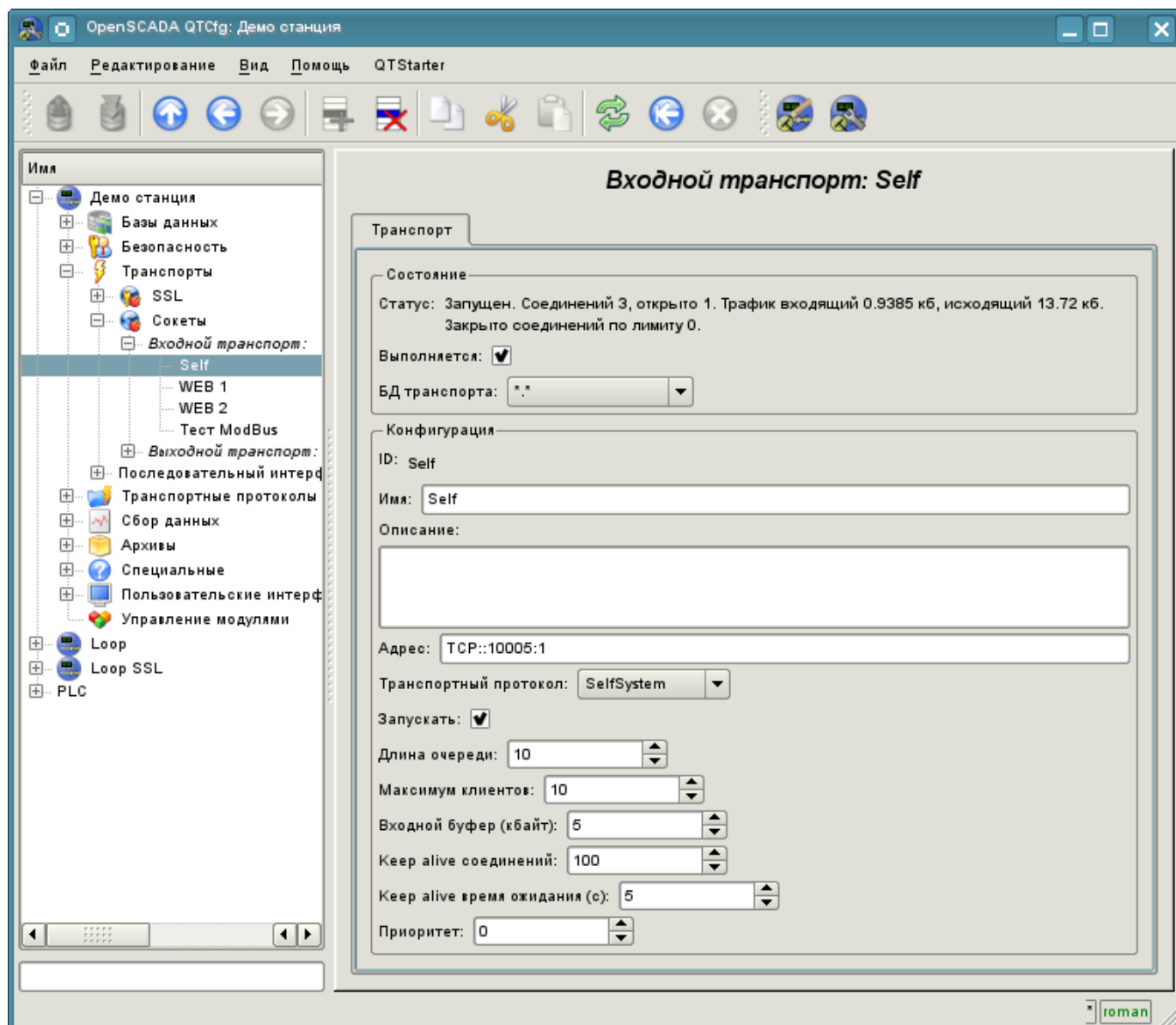


Рис.1. Диалог конфигурации входящего сокета.

С помощью этого диалога можно установить:

- Состояние транспорта, а именно: «Статус», «Запущен» и имя БД, содержащей конфигурацию.
- Идентификатор, имя и описание транспорта.
- Адрес транспорта. Формат адреса описан в таблице ниже.
- Выбор транспортного протокола.
- Состояние, в которое переводить контроллер при загрузке: «Запущен».
- Длина очереди сокетов, максимальное количество обслуживаемых клиентов и размер

входного буфера.

- Ограничения режима "Keep-alive" по количеству запросов и времени ожидания.
- Приоритет задач транспорта.

Особенности формирования адресов входящих сокетов приведены в таблице ниже:

Тип сокета	Адрес
TCP	<p><i>TCP:{адрес}:{порт}:{режим}</i></p> <p>где:</p> <ul style="list-style-type: none">• адрес – Адрес, на котором открывается сокет. Должен быть одним из адресов хоста. Если ничего не указано, то сокет будет доступен на всех интерфейсах хоста. Допускаются как символьное, так и IP представление адреса.• порт – Сетевой порт, на котором открывается сокет. Возможно указание символьного имени порта (в соответствии с /etc/services).• режим – режим работы входящего сокета (0 – разрывать соединение после сеанса приём-ответ; 1 – не разрывать). <p>Пример: "<i>TCP::10001:1</i>" — TCP-сокет доступен на всех интерфейсах, открыт на порту 10001 и соединения не разрывает.</p>
UDP	<p><i>UDP:{адрес}:{порт}</i></p> <p>где:</p> <ul style="list-style-type: none">• адрес – тоже что в TCP;• порт – тоже что в TCP. <p>Пример: "<i>UDP:localhost:10001</i>" — UDP-сокет доступен только на интерфейсе "localhost" и открыт на порту 10001.</p>
UNIX	<p><i>UNIX:{имя}:{режим}</i></p> <p>где:</p> <ul style="list-style-type: none">• имя – имя файла UNIX сокета;• режим – тоже что в TCP. <p>Пример: "<i>UNIX:/tmp/oscada:1</i>" — UNIX-сокет доступен через файл /tmp/oscada и соединения не разрывает.</p>

2. Исходящие транспорты

Сконфигурированный и запущенный исходящий транспорт открывает соединение с указанным сервером. При разрыве соединения исходящий транспорт отключается. Для возобновления соединения транспорт нужно снова запустить.

Главная вкладка страницы конфигурации исходящего сокета изображена на рис.2.

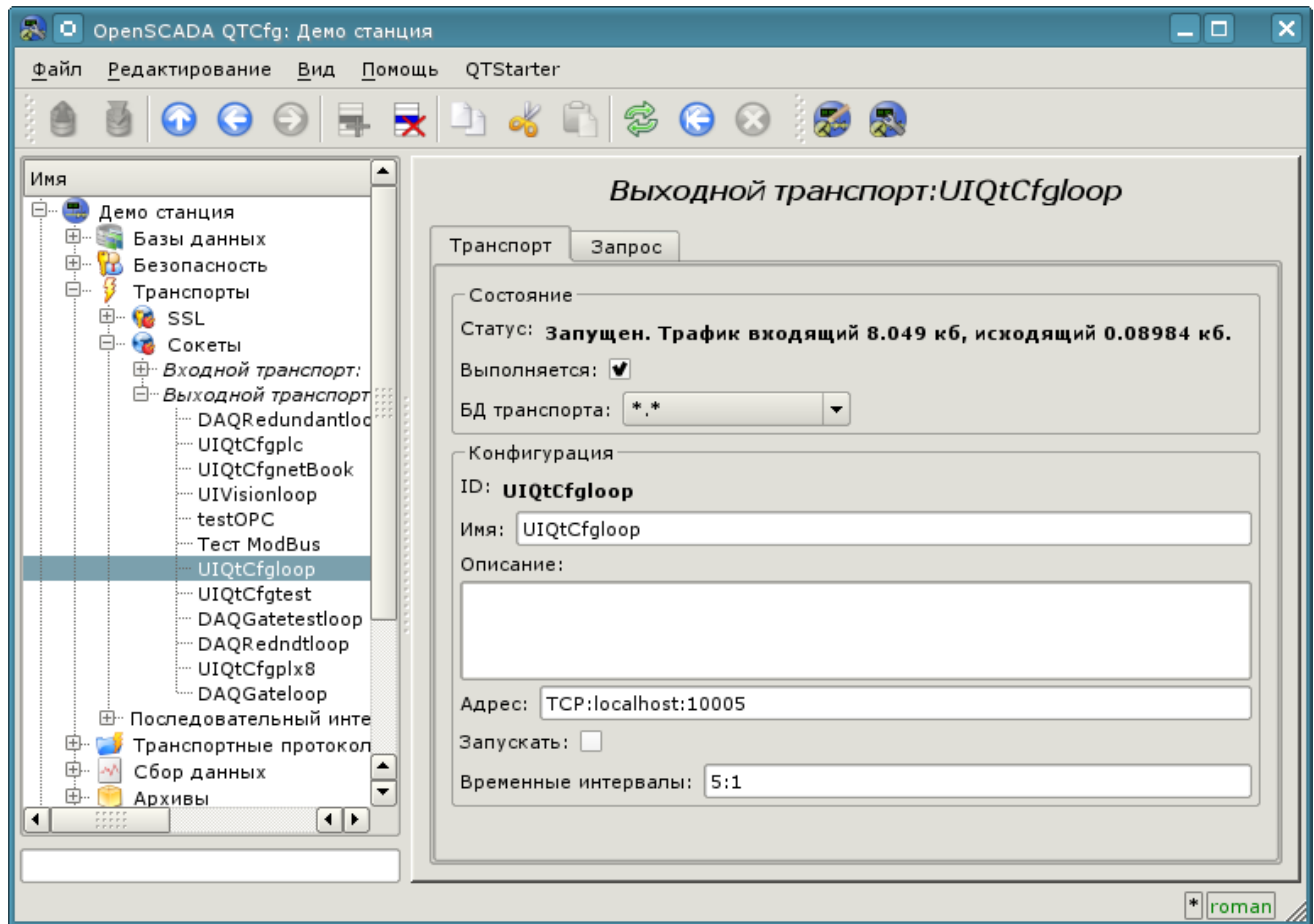


Рис.2. Главная вкладка страницы конфигурации исходящего сокета.

С помощью этого диалога можно установить:

- Состояние транспорта, а именно: "Статус", "Запущен" и имя БД, содержащей конфигурацию.
- Идентификатор, имя и описание транспорта.
- Адрес транспорта. Формат адреса описан в таблице ниже.
- Состояние, в которое переводить контроллер при загрузке: "Запущен".
- Временные интервалы соединения в формате: "**conn:next[:rep]**". Где:
 - *conn* — максимальное время ожидания ответа соединения, в секундах;
 - *next* — максимальное время ожидания продолжения ответа, в секундах;
 - *rep* — минимальный таймаут ожидания повтора запроса, в секундах.

Адреса исходящих сокетов различного типа формируются следующим образом:

Тип сокета	Адрес
TCP/UDP	<p><i>TCP:{адрес}:{порт}</i> <i>UDP:{адрес}:{порт}</i> где:</p> <ul style="list-style-type: none"> • адрес – Адрес, с которым выполняется соединение. Допускаются как символьное так и IP представление адреса. • порт – Сетевой порт, с которым выполняется соединение. Возможно указание символьного имени порта (в соответствии с /etc/services). <p>Пример: "<i>TCP:127.0.0.1:7634</i>" — соединится с портом 7634 на хосте 127.0.0.1.</p>
UNIX	<p><i>UNIX:{имя}</i> где:</p> <ul style="list-style-type: none"> • имя – имя файла UNIX сокета. <p>Пример: "<i>UNIX:/tmp/oscada</i>" — соединится с UNIX-сокетом через файл /tmp/oscada.</p>