

Модуль подсистемы “Транспорты” <Sockets>

Модуль:	Sockets
Имя:	Сокеты
Тип:	Транспорт
Источник:	tr_Sockets.so
Версия:	1.2.0
Автор:	Роман Савоченко
Описание:	Предоставляет транспорт основанный на сокетах. Поддерживаются интернет и unix сокеты. Интернет сокет использует TCP и UDP протоколы.
Лицензия:	GPL

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. Кроме индивидуальных настроек транспортов модуль предоставляет ещё три параметра общего назначения:

- длина очереди для TCP и UNIX сокетов;
- максимальное количество открытых клиентских TCP и UNIX сокетов;
- размер входного буфера.

Все эти параметры можно установить в конфигурационном файле и поменять из любого стандартного конфигуратора системы OpenSCADA.

1. Входящие транспорты

Сконфигурированный и запущенный входящий транспорт открывает серверный сокет для ожидания соединения клиента. В случае с UNIX сокетом создаётся файл UNIX сокета. Сокеты TCP и UNIX являются много-поточными, т.е. при подключении клиента к сокетам данных типов создаётся клиентский сокет и новый поток в котором и 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 раздробленных запросов.

Особенности формирования адресов входящих сокетов приведены в таблице ниже:

Тип сокета	Адрес
TCP	<i>TCP:[адрес]:[порт]:[режим]</i> где: <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). Пример: <i><TCP::10001:1></i> – TCP-сокет доступен на всех интерфейсах, открыт на порту 10001 и соединения не разрывает.
UDP	<i>UDP:[адрес]:[порт]</i> где: <ul style="list-style-type: none">• адрес – тоже что в TCP;• порт – тоже что в TCP. Пример: <i><UDP:localhost:10001></i> – UDP-сокет доступен только на интерфейсе “localhost” и открыт на порту 10001.
UNIX	<i>UNIX:[имя]:[режим]</i> где: <ul style="list-style-type: none">• имя – имя файла UNIX сокета;• режим – тоже что в TCP. Пример: <i><UNIX:/tmp/oscada:1></i> – UNIX-сокет доступен через файл /tmp/oscada и соединения не разрывает.

2. Исходящие транспорты

Сконфигурированный и запущенный исходящий транспорт открывает соединение с указанным сервером. Исходящие транспорты поддерживают восстановление соединения.

Адреса исходящих сокетов различного типа формируются следующим образом:

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