

# Модуль подсистемы “Транспорты” <Sockets>

<i>Модуль:</i>	Sockets
<i>Имя:</i>	Сокеты
<i>Тип:</i>	Транспорт
<i>Источник:</i>	tr_Sockets.so
<i>Версия:</i>	1.3.5
<i>Автор:</i>	Роман Савоченко
<i>Описание:</i>	Предоставляет транспорт, основанный на сокетах. Поддерживаются интернет и unix сокеты. Интернет сокет использует TCP и UDP протоколы.
<i>Лицензия:</i>	GPL

## Оглавление

<a href="#">Модуль подсистемы “Транспорты” &lt;Sockets&gt;</a> .....	1
<a href="#">Введение</a> .....	2
<a href="#">1 Входящие транспорты</a> .....	2
<a href="#">2 Исходящие транспорты</a> .....	3
<a href="#">3 Замечания</a> .....	3

## 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. 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 клиентов. В случае с UNIX сокетом, создаётся файл UNIX сокета. Сокеты TCP и UNIX являются много-поточными, т.е. при подключении клиента к сокетах 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 раздробленных, при передаче, запросов.

Особенности формирования адресов входящих сокетов приведены в таблице ниже:

Тип сокета	Адрес
TCP	<p><i>TCP:[адрес]:[порт]:[режим]</i> где:</p> <ul style="list-style-type: none"> <li>• 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.</li> <li>• порт – Сетевой порт, на котором открывается сокет. Возможно указание символьного имени порта (в соответствии с /etc/services).</li> <li>• режим – режим работы входящего сокета (0 – разрывать соединение после сеанса приём-ответ; 1 – не разрывать).</li> </ul> <p>Пример: <i>&lt;TCP::10001:1&gt;</i> – TCP-сокет доступен на всех интерфейсах, открыт на порту 10001 и соединения не разрывает.</p>
UDP	<p><i>UDP:[адрес]:[порт]</i> где:</p> <ul style="list-style-type: none"> <li>• адрес – тоже что в TCP;</li> <li>• порт – тоже что в TCP.</li> </ul> <p>Пример: <i>&lt;UDP:localhost:10001&gt;</i> – UDP-сокет доступен только на интерфейсе “localhost” и открыт на порту 10001.</p>

Тип сокета	Адрес
UNIX	<p><i>UNIX:[имя]:[режим]</i> где:</p> <ul style="list-style-type: none"> <li>• имя – имя файла UNIX сокета;</li> <li>• режим – тоже что в TCP.</li> </ul> <p>Пример: <i>&lt;UNIX:/tmp/oscada:1&gt;</i> – UNIX-сокеты доступны 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, по новой, включить.

Адреса исходящих сокетов различного типа формируются следующим образом:

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

## 3 Замечания

На данный момент транспорт поддерживает работу в классическом режиме передачи данных. Для повышения безопасности передачи данных, путём шифрования на уровне транспорта, запланировано добавление модуля транспорта SSL(Security Socket Layer).