

В новых версиях Modbus Universal MasterOPC и Multi-Protocol MasterOPC мы зашли на новый, только зарождающийся, рынок IoT (Internet of things – Интернет вещей), поддержав протокол MQTT. Кроме того, мы реализовали возможность выполнения групповых операций над устройствами и узлами.

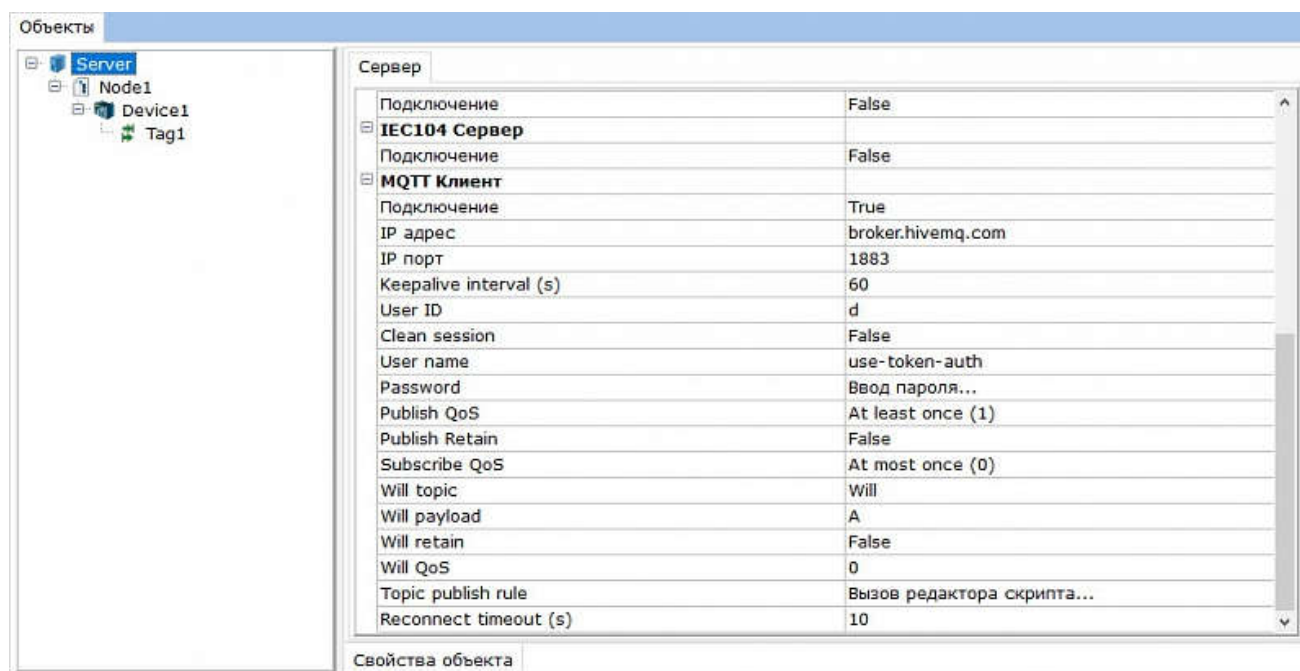
Начнем наш обзор с поддержки нового протокола MQTT. MQTT – это открытый упрощенный протокол работающих в сетях TCP/IP. Основное назначение протокола MQTT это различные элементы для систем умного дома – выключатели, датчики температуры, освещенности и т.д. Помимо этого протокол MQTT активно применяется как механизм передачи данных в различных облачных сервисах, например, Microsoft Azure.

Его особенностью является обмен по принципу издатель-подписчик с промежуточным сервером, называемым «брокером». Протокол прост в использовании и имеет множество поддерживаемых устройств, кроме того поскольку в основу протокола заложен режим подписки (то есть передачи данных по определенному событию или изменению значения) это снижает нагрузку на сеть и вычислительные мощности устройства. Протокол прост в администрировании, за счет разных моделей подтверждения получения может работать в том числе на нестабильных каналах связи, а также имеет возможности шифрования данных.

В нашем OPC сервере мы поддерживали MQTT клиент, что позволяет интегрировать различные устройства в SCADA, а также работать в обратном направлении – передавать данные от различных протоколов в облачные сервисы.

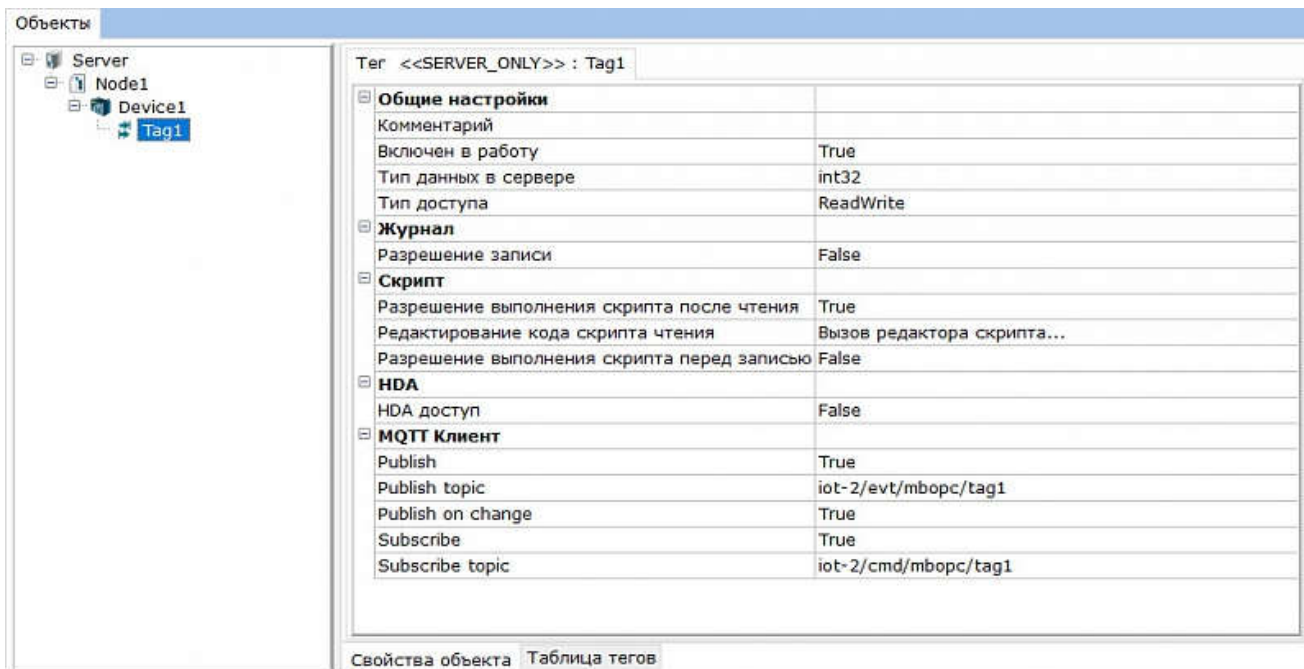
Как же настроить MQTT клиент? Рассмотрим, в качестве примера, настройку в Modbus Universal MasterOPC сервере. Сначала необходимо сделать MQTT сервер (брокер). Можно развернуть его локально на этой машине (например, используя сервер [Mosquitto](#)), подключиться к уже готовому, или использовать любой облачный MQTT сервер – например [www.hivemq.com](#)

Затем, в OPC сервере нужно включить режим MQTT клиента.



Теперь необходимо выполнить его настройку – указать IP адрес брокера, логин и пароль. Если планируется что-то передавать из сервера, то можно изменить скрипт Topic publish rule – с его помощью можно кастомизировать строку, формируемую серверу при посылке данных (Publish), например, можно указать метку времени или признак качества, произвольно обработать отправляемые данные и т.д.

После включения MQTT клиента, у каждого тега появляется возможность настроить его на прием или передачу MQTT сообщений. Для этого предназначены настройки Publish (передача) и Subscribe (подписка-получение). Включите нужный режим и укажите имя топики – в случае публикации, сообщение будет посылаться с этим идентификатором, и все устройства, подписанные на этот идентификатор, получат данные. При подписке – в тег будут записываться только данные полученные от устройств с этим идентификатором.



В поставку сервера входит конфигурация MQTT_Demo в которой происходит пересылка изменяющегося значения, а также прием данных от других устройств через облачный брокер hivemq.com. Для проверки работы конфигурации можно подключиться к данному брокеру с тем же логином и посмотреть данные передаваемые сервером. В качестве такого клиента можно использовать браузерное расширение [MQTTLens](#).

Протокол MQTT будет доступен в Modbus Universal MasterOPC сервере, а также в Multi-Protocol MasterOPC сервере. В Modbus Universal MasterOPC функционал доступен в версиях на 10, 60 и 200 тысяч точек, а также в Trial версии. В Multi-Protocol MasterOPC сервер он доступ при наличии опции [MQTT-Client](#).