



**Руководство пользователя
Драйвер Vasnet2mqtt**

Оглавление

1. Ключевые функции	3
2. Функционал и реализация	3
2.1 Ключи запуска приложения	3
3. Описание файлов конфигурации	4
3.1 Описание корневого файла конфигурации	4
3.2 Описание файла конфигурации bacnet-устройств	7
3.2.1 Раздел Device	8
3.2.2 Раздел data_model	8
3.2.3 Поддерживаемые драйвером типы bacnet-объектов	10
3.2.4 Поддерживаемые драйвером свойства bacnet-объектов	11
3.2.5 Поддерживаемые драйвером классы ошибок	12
3.2.6 Поддерживаемые драйвером классы ошибок	12
3.2.7 Поддерживаемые драйвером классы ошибок	12
3.3 Описание правил дополнительного модуля валидации	15
4. Примеры файлов конфигурации YAML	16
4.1 Bacnet.yaml	16
4.2 bacnet_device1.yaml	17
4.3 bacnet_device2.yaml	18
5. BScan	19
5.1 Команда getinfo	19
5.2 Команда whois	21

BACnet2MQTT — приложение, предназначенное для взаимодействия с устройствами BACnet/IP. Оно позволяет собирать данные с BACnet-устройств, публиковать их в брокере MQTT и форматировать полученные данные. Приложение предназначено для использования в системах, где необходима интеграция различных устройств BACnet с брокером MQTT для обработки данных в реальном времени.

Приложение представляет собой бинарный файл под Linux той или иной архитектуры процессора. Может поставляться в виде deb-пакета. Конфигурирование производится через yaml-файлы конфигурации.

1. Ключевые функции

Приложение поддерживает следующие функции:

- Периодическое чтение данных из bacnet-устройств;
 - Отправка считанных данных в виде mqtt-топиков в локальный MQTT-брокер;
 - Управление bacnet-устройством через управляющие mqtt-топики
 - Запись данных в bacnet-устройства из полученных mqtt-топиков;
- Гибкое конфигурирование посредством YAML-файлов

2. Функционал и реализация

2.1 Ключи запуска приложения

Для вызова справки по ключам (аргументам командной строки), поддерживаемым приложением можно использовать **--help**:

```
andromeda-user@host-linux:~$ ./bacnet2mqtt_x.x.x.x_linux_amd64 --help
Usage of ./bacnet2mqtt_x.x.x.x_linux_amd64:
-c, --cfgpath string      Путь до файла конфигурации (default "configs/bacnet2mqtt/bacnet.yaml")
-v, --version             Вывести версию драйвера Bacnet2MQTT
-l, --level_log int      Уровень логирования драйвера (1-info, 2-warn, 3-debug, 4-error) (default 1)
-w, --write_in_logfile   Запись логов в файл (true / false)
-p, --path_log string    Путь до файла логов
--unsafe                 Отключение валидации конфигураций входных/выходных топиков
```

Если ключ **-c, --cfgpath** не будет передан при запуске, то драйвер будет искать файл конфигурации в пути по умолчанию "configs/bacnet2mqtt/bacnet.yaml".

Ключи, связанные с логами, могут быть заданы в файле конфигурации, который передается через ключи **-c, --cfgpath**.

ВАЖНО!

Настройки применяются в порядке приоритета (от наивысшего к низшему):

1. Аргумент командной строки;
2. Параметр в файле конфигурации;
3. Значение по умолчанию.

3. Описание файлов конфигурации

Конфигурация драйвера задается через параметры командной строки, как описано выше, и в нескольких yaml-файлах: один корневой и несколько файлов конфигурации конкретных bacnet-устройств.

Корневой файл конфигурации содержит общие настройки драйвера, такие как:

- Логирование;
- Прослушиваемый udp-порт;
- Адрес mqtt-брокера;
- Пути к файлам конфигурации опрашиваемых bacnet-устройств. Путь к ним указывается через ключ `-c`, `--cfgpath`. Файлы конфигурации bacnet-устройств содержат настройки, необходимые для их опроса и управления, подробнее в п.3.2.2.

3.1 Описание корневого файла конфигурации

Содержимое файла конфигурации выглядит следующим образом:

```
bacnet:
  enabled: true # Флаг включения/отключения работы драйвера
  udp:
    # iface: lo # Интерфейс, на котором будет запущен UDP-listener
    port: 47808 # Порт, на котором будет запущен UDP-listener
    ip: 0.0.0.0/24 # ip, на котором будет запущен UDP-listener и маска подсети
  mqtt:
    broker: localhost:1883 # Или tcp://localhost:1883 или unix:///var/run/mosquitto/mosquitto.sock
    client_id: bacnet2mqtt
    keepalive: 60

  logger:
    write_in_logfile: false
    level_log: 3
    path_log: bacnet2mqtt.log
    max_size_log: 5
    max_backups_log: 70
    max_age_log: 30

  devices:
    bacnet_device1:
      file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device1.yaml
    bacnet_device2:
      file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device2.yaml
    bacnet_device3:
      file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device3.yaml
    bacnet_device4:
      file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device4.yaml
    bacnet_device5:
      file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device5.yaml
```

Раздел `udp` описывает сетевые настройки, с которыми запускается драйвер.

ВАЖНО!

При одновременном указании `iface` и `ip` приоритет остается за `iface`

Раздел `mqtt` описывает настройки подключения к mqtt-брокеру. Все поля являются необязательными и могут не указываться, в таком случае будут приняты значения по умолчанию.

Раздел `logger` описывает настройки логирования работы приложения. Все поля являются необязательными и могут не указываться, в таком случае будут приняты значения по умолчанию.

ВАЖНО!

Настройки параметров `write_in_logfile`, `level_log` и `path_log` доступны через аргументы командной строки.

Раздел `devices` описывает условные имена опрашиваемых bacnet-устройств и пути к их файлам конфигурации. Имена устройств должны быть уникальными.

Описание в YAML	Название	Обязательный / Необязательный параметр	Описание	Допустимые значения	Значения по умолчанию
<code>enabled</code>	Режим работы	Обязательный	Служит для включения, отключения работы драйвера	<code>true, false</code>	—
<code>port</code>	Порт	Необязательный	UDP-порт, на котором будет запущен драйвер. С этого порта будут отправляться запросы на чтение и приниматься ответы и широковещательные BACnet-сообщения.	1-65535	47808
<code>iface</code>	Сетевой интерфейс	Необязательный	Имя сетевого интерфейса, на котором будет запущен UDP-листнер. Драйвер привязывается к первому IP указанного интерфейса. Маска подсети определяется из конфигурации интерфейса. При одновременном указании <code>iface</code> и <code>ip</code> приоритет остаётся за <code>iface</code> .	—	""
<code>ip</code>	IP-адрес	Необязательный	Конкретный IP-адрес, на котором драйвер будет слушать unicast-сообщения. При значении 0.0.0.0 драйвер слушает все интерфейсы. Маска подсети используется при работе с broadcast-сообщениями.	—	0.0.0.0/0
<code>broker</code>	MQTT-брокер	Необязательный	Адрес MQTT-брокера, на который будут отправляться топики с данными из BACnet-устройств.	Примеры: <code>tcp://localhost:1883</code> , <code>unix:///var/run/mosquitto/mosquitto.sock</code> . DNS-имена не поддерживаются, кроме <code>localhost</code> .	127.0.0.1:1883
<code>client_id</code>	Идентификатор клиента	Необязательный	Значение <code>client_id</code> , с которым драйвер подключается к MQTT-брокеру.	—	<code>bacnet2mqtt</code>
<code>keepalive</code>	Время жизни соединения	Необязательный	Время жизни MQTT-соединения в секундах без передачи данных.	1-360	60
<code>write_in_logfile</code>	Запись в лог-файл	Необязательный	Включение записи логов в файл.	<code>true, false</code>	<code>false</code>

level_log	Уровень логирования	Необязательный	Уровень логирования.	1-4 (1-info, 2-warn, 3-debug, 4-error).	1
path_log	Путь к лог-файлу	Обязателен при write_in_logfile = true.	Путь к файлу логов.	—	—
max_size_log	Размер лог-файла	Необязательный	Максимальный размер лог-файла в мегабайтах.	1-5	5
max_backups_log	Количество резервных копий	Необязательный	Максимальное количество резервных копий логов.	1-100	100
max_age_log	Возраст логов	Необязательный	Максимальный возраст лог-файлов в днях.	1-30	30
devices	Конфигурация устройств	Обязательный	Описывает условные имена опрашиваемых VASnet-устройств и пути к их файлам конфигурации. Имена устройств должны быть уникальными.	—	—
enabled	Режим работы	Обязательный	Служит для включения, отключения работы драйвера	true, false	—

3.2 Описание файла конфигурации баснет-устройств

Содержимое файла конфигурации выглядит следующим образом:

```
device:
  name: bacnet_device1
  enabled: true
  root_topic_mqtt: room1111
  address: 172.26.112.1:54502
  # timeout_s: 5
  # max_apdu: 1476
  # max_quantity: 100
  data_model:
    read:
      - read_period_ms: 1000
        parameters:
          - parameter_name: Indoor Temperature
            precision: 2
            transform: 0.1:0
            object:
              object_type: AnalogInput
              instance: 0
              property_id: PresentValue
          - parameter_name: Glycol Water Temperature
            precision: 2
            object:
              object_type: AnalogInput
              instance: 1
              property_id: PresentValue
          - parameter_name: Outdoor Temperature
            precision: 2
            object:
              object_type: AnalogInput
              instance: 2
              property_id: PresentValue
        - read_period_ms: 2000
          parameters:
            - parameter_name: Effective SetPoint
              precision: 2
              transform: 0.1:0
              object:
                object_type: AnalogValue
                instance: 0
                property_id: PresentValue
            - parameter_name: Setoint 1
              precision: 2
              object:
                object_type: AnalogValue
                instance: 1
                property_id: PresentValue
            - parameter_name: Setoint 2
              precision: 2
              object:
                object_type: AnalogValue
                instance: 2
                property_id: PresentValue
    write:
      - parameter_name: Indoor Temperature
        command_topic: on
        object:
          object_type: AnalogInput
          instance: 0
      - parameter_name: Setoint 1
        command_topic: on
        object:
          object_type: AnalogValue
          instance: 1
```

3.2.1 Раздел Device

Содержит общие настройки устройства:

Описание в YAML	Название	Обязательный / Необязательный параметр	Описание	Допустимые значения	Значения по умолчанию
name	Имя устройства	Обязательный	Имя устройства, используемое в логах для идентификации сообщений, связанных с этим устройством.	—	—
enabled	Включено	Обязательный	Флаг включения/отключения опроса устройства. Поле является обязательным. Допустимые значения: true, false.	true, false	—
root_topic_mqtt	Корневой MQTT-топик	Обязательный	Часть полного MQTT-топика в структуре /devices/\${root_topic_mqtt}/controls/\${parameter_name}.	Не должно содержать символы + и #	—
address	Адрес устройства	Обязательный	IP-адрес и порт ВАСnet-устройства.	Формат: 192.168.57.1 00:47808.	—
timeout_s	Таймаут ожидания ответа	Необязательный	Время ожидания ответа от ВАСnet-устройства (в секундах).	1-5	5
max_apdu	Максимальный размер APDU	Необязательный	Максимальный размер APDU-части пакета ВАСnet (в байтах). Для ВАСnet/IP максимальное значение — 1476. Не рекомендуется изменять без необходимости.	1-1476	1476
max_quantity	Максимальное количество объектов	Необязательный	Максимальное количество объектов в одном запросе.	1-150	100
name	Имя устройства	Обязательный	Имя устройства, используемое в логах для идентификации сообщений, связанных с этим устройством.	—	—
enabled	Включено	Обязательный	Флаг включения/отключения опроса устройства. Поле является обязательным. Допустимые значения: true, false.	true, false	—
root_topic_mqtt	Корневой MQTT-топик	Обязательный	Часть полного MQTT-топика в структуре /devices/\${root_topic_mqtt}/controls/\${parameter_name}.	Не должно содержать символы + и #	—
address	Адрес устройства	Обязательный	IP-адрес и порт ВАСnet-устройства.	Формат: 192.168.57.1 00:47808.	—
timeout_s	Таймаут ожидания ответа	Необязательный	Время ожидания ответа от ВАСnet-устройства (в секундах).	1-5	5
max_apdu	Максимальный размер APDU	Необязательный	Максимальный размер APDU-части пакета ВАСnet (в байтах). Для ВАСnet/IP максимальное значение — 1476. Не рекомендуется изменять без необходимости.	1-1476	1476
max_quantity	Максимальное количество объектов	Необязательный	Максимальное количество объектов в одном запросе.	1-150	100
name	Имя устройства	Обязательный	Имя устройства, используемое в логах для идентификации сообщений, связанных с этим устройством.	—	—

3.2.2 Раздел data_model

Содержит в себе два подраздела read (для описания баснет-параметров на чтение) и write (для описания баснет-параметров на запись):

Раздел **read** структурно представляет собой группы сигналов, объединенные по значению их периода опроса, т.е. одна группа сигналов опрашивается по периоду 1000ms, вторая группа по периоду 2000ms. Должна быть указана минимум одна группа опроса. Каждая группа имеет следующие поля:

Описание в YAML	Название	Обязательный / Необязательный параметр	Описание	Допустимые значения	Значения по умолчанию
read_period_ms	Период опроса	Обязательный	Период опроса устройства в миллисекундах.	Целое число > 0	—
parameter_name	Имя параметра	Обязательный	Будет использоваться как <code>/devices/\${root_topic_mqtt}/controls/\${parameter_name}</code> . Если в названии параметра указан <code>/devices/ ...</code> то для публикации не учитывается преобразование с <code>\${root_topic_mqtt}</code> и берётся полное имя <code>parameter_name</code> .	Будет использоваться как	Будет использоваться как
precision	Точность округления	Необязательный	Кол-во знаков после запятой; Не применяется к строковым и булевым типам данных. Значение '-1' отключает функцию округления. Значение будет передано с тем количеством знаков, с каким получено из сервера.	-1:10	-1
transform	Формула преобразования	Необязательный	Формула для преобразования значения $y = (a * x + b)$; x - исходное значение, y - преобразованное значение для записи, по умолчанию $(a:b) = (1:0)$	—	1:0
object	ВАСnet-объект	Обязательный	Содержит информацию о ВАСnet-объекте, к которому относится параметр.	—	—
object_type	Тип объекта	Обязательный	Тип ВАСnet-объекта.	Поддерживаемые типы приведены в таблице 1	—
instance	Экземпляр объекта	Обязательный	Экземпляр ВАСnet-объекта указанного типа.	Целое число ≥ 0	—
property_id	Идентификатор свойства	Обязательный	Идентификатор свойства ВАСnet-объекта.	Поддерживаемые идентификаторы свойств приведены в таблице 2	—

Раздел **write** представлен как список параметров баснет-устройства с дополнительным полем `command_topic`.

Общий принцип, следующий на основе параметров, приведенных в `root_topic_mqtt`, `command_topic` и `parameter_name` формируется топик управления `'devices/${root_topic_mqtt}/controls/${parameter_name}/${command_topic}'`, на который подписывается драйвер на mqtt-брокере. При получении этого топика от брокера, его значение `payload` записывается в соответствующее свойство баснет-объекта:

Описание в YAML	Название	Обязательный / Необязательный параметр	Описание	Допустимые значения	Значения по умолчанию
parameter_name	Имя параметра	Обязательный	Имя параметра. Используется в структуре топика <code>/devices/\${root_topic_mqtt}/controls/\${parameter_name}/\${command_topic}</code> . Если значение <code>parameter_name</code> начинается с <code>/devices/</code> , то <code>root_topic_mqtt</code> не используется, а итоговый	Не должно содержать символы + и #	—

			топик формируется как $\${parameter_name}/\${command_topic}$.		
command_topic	Суффикс топика управления	Необязательный	Значение, используемое в структуре управляющих топиков $/devices/\${root_topic_mqtt}/controls/\${parameter_name}/\${command_topic}$.	Не должно содержать символы + и #	оп
object	ВАСnet-объект		Содержит ВАСnet-информацию о параметре.	—	—
object_type	Тип ВАСnet-объекта	Обязательный	Тип ВАСnet-объекта. Поддерживаемые типы приведены в таблице 1.	AnalogOutput, AnalogValue, BinaryOutput, BinaryValue, MultiStateOutput, MultiStateValue, CharacterString	—
instance	Экземпляр ВАСnet-объекта	Обязательный	Экземпляр объекта указанного типа. Поле является обязательным. Допустимые значения: целое число ≥ 0 .	Целое число ≥ 0	—

ВАЖНО!

В драйвере *basnet2mqtt* запись в бакнет-объекты возможна только в свойство *PresentValue*, поэтому параметр *property_id* в разделе *Write* не работает. Запись автоматически будет выполняться только в свойство *PresentValue* объекта, в отличие от раздела *Read*, где из бакнет объекта можно читать не только *PresentValue*.

3.2.3 Поддерживаемые драйвером типы баснет-объектов

Строковое представление object_type	Числовое представление object_type	Описание	Допустимые баснет-устройства
AnalogInput	0	Неуправляемый аналоговый вход	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, Units
AnalogOutput	1	Управляемый аналоговый выход	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, Units
AnalogValue	2	Управляемое аналоговое значение	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, Units
BinaryInput	3	Неуправляемый дискретный вход	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, ActiveText, InactiveText, Polarity
BinaryOutput	4	Управляемый дискретный выход	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, ActiveText, InactiveText, Polarity
BinaryValue	5	Управляемое дискретное значение	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, ActiveText, InactiveText, Polarity
Device	8	ВАСnet-Объект, содержащий основную информацию об устройстве	Description, ObjectName, ObjectIdentifier, ObjectType,

			ObjectList, ModelName, VendorName,
MultiStateInput	13	Неуправляемый BACnet-объект, который представляет собой дискретную переменную с несколькими предопределёнными состояниями. В отличие от бинарных объектов (BinaryInput/Output/Value), которые имеют только два состояния, поддерживает множество значений, каждое из которых имеет своё смысловое описание	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, NumberOfStates, StateText
MultiStateOutput	14	BACnet-объект, аналогичный MultiStateInput, но управляемый	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, NumberOfStates, StateText
MultiStateValue	19	BACnet-объект, аналогичный MultiStateInput, но управляемый. Представляет собой уставку чего-либо.	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue, NumberOfStates, StateText
CharacterString	40	Изменяемый BACnet-объект, содержащий текстовую информацию.	Description, ObjectName, ObjectIdentifier, ObjectType, PresentValue,

3.2.4 Поддерживаемые драйвером свойства bacnet-объектов

Строковое представление property_id	Числовое представление property_id	Описание	Допустимые bacnet-устройства
Description	28	Стандартное строковое свойство, содержащее описание данного BACnet-объекта	AnalogInput, AnalogOutput, AnalogValue, BinaryInput, BinaryOutput, BinaryValue, MultiStateInput, MultiStateOutput, MultiStateValue, CharacterString, Device
ModelName	70	Имя устройства. Применимо только к BACnet-объекту типа Device	Device
VendorName	121	Указывает имя компании-производителя BACnet-устройства. Обычно берется из BACnet Vendor ID, зарегистрированного в ASHRAE (ассоциация, поддерживающая стандарт BACnet).	Device
ObjectList	76	Список объектов в устройстве. Применимо только к BACnet-объекту типа Device	Device
ObjectName	77	Стандартное строковое свойство BACnet-объектов, содержащее уникальное читаемое имя объекта	AnalogInput, AnalogOutput, AnalogValue, BinaryInput, BinaryOutput, BinaryValue, MultiStateInput, MultiStateOutput, MultiStateValue, CharacterString, Device
ObjectIdentifier	75	Идентификатор бакнет-объекта. Содержащий в себе тип объекта (AnalogInput, AnalogOutput и т.д.) и номер экземпляра	AnalogInput, AnalogOutput, AnalogValue, BinaryInput, BinaryOutput, BinaryValue, MultiStateInput, MultiStateOutput, MultiStateValue, CharacterString, Device
ObjectType	79	Свойство бакнет объекта, описывающее его тип (AnalogInput, AnalogOutput и т.д.)	AnalogInput, AnalogOutput, AnalogValue, BinaryInput, BinaryOutput, BinaryValue, MultiStateInput, MultiStateOutput, MultiStateValue, CharacterString, Device
PresentValue	85	Свойство BACnet-объектов, содержащее актуальное состояние или измеренное значение объекта на текущий момент. Оно является основным для чтения данных и управления в BACnet-системах.	AnalogInput, AnalogOutput, AnalogValue, BinaryInput, BinaryOutput, BinaryValue, MultiStateInput, MultiStateOutput, MultiStateValue, CharacterString

Units	117	Единицы измерения значения в PresentValue данного ВАСnet-объекта.	AnalogInput, AnalogOutput, AnalogValue
ActiveText	4	Это свойство определяет текстовое состояние объекта при значении PresentValue объекта равным True/1 (например, включен, Авария, состояние "1" и т. д.).	BinaryInput, BinaryOutput, BinaryValue
InactiveText	46	Это свойство определяет текстовое состояние объекта при значении PresentValue объекта равным False/0 (например, выключен, Норма, состояние "0" и т. д.).	BinaryInput, BinaryOutput, BinaryValue
Polarity	84	Это свойство определяет применение инверсии к значению PresentValue.	BinaryInput, BinaryOutput, BinaryValue
NumberOfStates	74	Количество возможных состояний Enum объектов в протоколе ВАСnet.	MultiStateInput, MultiStateOutput, MultiStateValue
StateText	110	Описаний состояний Enum объектов в протоколе ВАСnet.	MultiStateInput, MultiStateOutput, MultiStateValue

ВАЖНО!

На данный момент в файле конфигурации можно указывать только строковое представление *object_type* и *property_id* (*AnalogInput* вместо 0, *PresentValue* вместо 85).

3.2.5 Поддерживаемые драйвером классы ошибок

Числовое представление класса ошибки	Строковое представление класса ошибки	Описание
0x00 (0)	DeviceError	Ошибки, связанные с устройством (недоступность, перегрузка, сбой).
0x01 (1)	ObjectError	Ошибки, связанные с объектами ВАСnet (несуществующий объект, недоступность).
0x02 (2)	PropertyError	Ошибки, связанные со свойствами объекта (неподдерживаемое свойство, неверное значение).
0x03 (3)	ResourcesError	Ошибки нехватки ресурсов (память, процессорное время).
0x04 (4)	SecurityError	Ошибки безопасности (неавторизованный доступ, нарушение политик).
0x05 (5)	ServicesError	Ошибки выполнения сервисов ВАСnet (неподдерживаемый сервис, неверные параметры).
0x06 (6)	VTErrror	Ошибки, связанные с виртуальными терминальными сессиями (неподдерживаемые настройки, попытка записи в закрытую сессию)
0x07 (7)	CommunicationError	Ошибки связи (таймауты, разрыв соединения, проблемы с маршрутизацией).

3.2.6 Поддерживаемые драйвером классы ошибок

Числовое представление класса ошибки	Строковое представление класса ошибки	Описание
0x00 (0)	DeviceError	Ошибки, связанные с устройством (недоступность, перегрузка, сбой).
0x01 (1)	ObjectError	Ошибки, связанные с объектами ВАСnet (несуществующий объект, недоступность).
0x02 (2)	PropertyError	Ошибки, связанные со свойствами объекта (неподдерживаемое свойство, неверное значение).
0x03 (3)	ResourcesError	Ошибки нехватки ресурсов (память, процессорное время).
0x04 (4)	SecurityError	Ошибки безопасности (неавторизованный доступ, нарушение политик).
0x05 (5)	ServicesError	Ошибки выполнения сервисов ВАСnet (неподдерживаемый сервис, неверные параметры).
0x06 (6)	VTErrror	Ошибки, связанные с виртуальными терминальными сессиями (неподдерживаемые настройки, попытка записи в закрытую сессию)
0x07 (7)	CommunicationError	Ошибки связи (таймауты, разрыв соединения, проблемы с маршрутизацией).

3.2.7 Поддерживаемые драйвером классы ошибок

Числовое представление класса ошибки	Строковое представление класса ошибки	Описание
0x00 (0)	Other	Другая (неспецифицированная) ошибка
0x01 (1)	AuthenticationFailed	Ошибка аутентификации
0x02 (2)	ConfigurationInProgress	Идёт процесс конфигурации устройства
0x03 (3)	DeviceBusy	Устройство занято и не может обработать запрос
0x04 (4)	DynamicCreationNotSupported	Динамическое создание объектов не поддерживается
0x05 (5)	FileAccessDenied	Доступ к файлу запрещён
0x06 (6)	IncompatibleSecurityLevels	Несовместимые уровни безопасности между устройствами
0x07 (7)	InconsistentParameters	Несогласованные параметры в запросе
0x08 (8)	InconsistentSelectionCriterion	Несогласованный критерий выбора объектов
0x09 (9)	InvalidDataType	Недопустимый тип данных
0x0A (10)	InvalidFileAccessMethod	Недопустимый метод доступа к файлу
0x0B (11)	InvalidFileStartPosition	Недопустимая стартовая позиция в файле
0x0C (12)	InvalidOperatorName	Недопустимое имя оператора
0x0D (13)	InvalidParameterDataType	Недопустимый тип данных параметра
0x0E (14)	InvalidTimeStamp	Некорректная временная метка
0x0F (15)	KeyGenerationError	Ошибка генерации ключа шифрования
0x10 (16)	MissingRequiredParameter	Отсутствует обязательный параметр
0x11 (17)	NoObjectsOfSpecifiedType	Нет объектов указанного типа
0x12 (18)	NoSpaceForObject	Недостаточно памяти для создания объекта
0x13 (19)	NoSpaceToAddListElement	Недостаточно памяти для добавления элемента в список
0x14 (20)	NoSpaceToWriteProperty	Недостаточно памяти для записи свойства
0x15 (21)	NoVtSessionsAvailable	Нет доступных сессий виртуального терминала
0x16 (22)	PropertyIsNotAList	Свойство не является списком
0x17 (23)	ObjectDeletionNotPermitted	Удаление объекта не разрешено
0x18 (24)	ObjectIdentifierAlreadyExists	Идентификатор объекта уже существует
0x19 (25)	OperationalProblem	Операционная проблема (внутренний сбой устройства)
0x1A (26)	PasswordFailure	Ошибка пароля (неверный пароль)
0x1B (27)	ReadAccessDenied	Доступ на чтение запрещён
0x1C (28)	SecurityNotSupported	Запрошенный уровень безопасности не поддерживается
0x1D (29)	ServiceRequestDenied	Запрос сервиса отклонён
0x1E (30)	Timeout	Таймаут операции
0x1F (31)	UnknownObject	Неизвестный объект
0x20 (32)	UnknownProperty	Неизвестное свойство
0x22 (34)	UnknownVtClass	Неизвестный класс виртуального терминала
0x23 (35)	UnknownVtSession	Неизвестная сессия виртуального терминала
0x24 (36)	UnsupportedObjectType	Неподдерживаемый тип объекта
0x25 (37)	ValueOutOfRange	Значение вне допустимого диапазона
0x26 (38)	VtSessionAlreadyClosed	Сессия виртуального терминала уже закрыта
0x27 (39)	VtSessionTerminationFailure	Ошибка завершения сессии виртуального терминала
0x28 (40)	WriteAccessDenied	Доступ на запись запрещён
0x29 (41)	CharacterSetNotSupported	Неподдерживаемый набор символов
0x2A (42)	InvalidArrayIndex	Недопустимый индекс массива
0x2B (43)	CovSubscriptionFailed	Ошибка подписки на изменение значения (COV)
0x2C (44)	NotCovProperty	Свойство не поддерживает уведомления COV
0x2D (45)	OptionalFunctionalityNotSupported	Опциональная функциональность не поддерживается
0x2E (46)	InvalidConfigurationData	Некорректные данные конфигурации
0x2F (47)	DatatypeNotSupported	Тип данных не поддерживается
0x30 (48)	DuplicateName	Дублирующееся имя объекта
0x31 (49)	DuplicateObjectID	Дублирующийся идентификатор объекта
0x32 (50)	PropertyIsNotAnArray	Свойство не является массивом
0x33 (51)	AbortBufferOverflow	Прерывание: переполнение буфера
0x34 (52)	AbortInvalidApdulnThisState	Прерывание: недопустимый APDU в текущем состоянии
0x35 (53)	AbortPreemptedByHigherPriorityTask	Прерывание: вытеснение задачей с высоким приоритетом
0x36 (54)	AbortSegmentationNotSupported	Прерывание: сегментация не поддерживается
0x37 (55)	AbortProprietary	Прерывание: проприетарная причина
0x38 (56)	AbortOther	Прерывание: другая причина
0x39 (57)	InvalidTag	Недопустимый тег в APDU
0x3A (58)	NetworkDown	Сеть недоступна
0x3B (59)	RejectBufferOverflow	Отклонение: переполнение буфера

0x3C (60)	RejectInconsistentParameters	Отклонение: несогласованные параметры
0x3D (61)	RejectInvalidParameterDataType	Отклонение: недопустимый тип данных параметра
0x3E (62)	RejectInvalidTag	Отклонение: недопустимый тег
0x3F (63)	RejectMissingRequiredParameter	Отклонение: отсутствует обязательный параметр
0x40 (64)	RejectParameterOutOfRange	Отклонение: параметр вне допустимого диапазона
0x41 (65)	RejectTooManyArguments	Отклонение: слишком много аргументов
0x42 (66)	RejectUndefinedEnumeration	Отклонение: неопределённое значение перечисления
0x43 (67)	RejectUnrecognizedService	Отклонение: неизвестный сервис
0x44 (68)	RejectProprietary	Отклонение: проприетарная причина
0x45 (69)	RejectOther	Отклонение: другая причина
0x46 (70)	UnknownDevice	Неизвестное устройство
0x47 (71)	UnknownRoute	Неизвестный маршрут
0x48 (72)	ValueNotInitialized	Значение не инициализировано
0x49 (73)	InvalidEventState	Недопустимое состояние события
0x4A (74)	NoAlarmConfigured	Нет настроенной тревоги
0x4B (75)	LogBufferFull	Буфер журнала заполнен
0x4C (76)	LoggedValuePurged	Значение из журнала удалено
0x4D (77)	NoPropertySpecified	Свойство не указано
0x4E (78)	NotConfiguredForTriggeredLogging	Не настроено для триггерного логирования
0x4F (79)	UnknownSubscription	Неизвестная подписка
0x50 (80)	ParameterOutOfRange	Параметр вне допустимого диапазона
0x51 (81)	ListElementNotFound	Элемент списка не найден
0x52 (82)	Busy	Устройство занято
0x53 (83)	CommunicationDisabled	Связь отключена
0x54 (84)	Success	Успешное выполнение (не ошибка)
0x55 (85)	AccessDenied	Доступ запрещён
0x56 (86)	BadDestinationAddress	Недопустимый адрес назначения
0x57 (87)	BadDestinationDeviceID	Недопустимый ID устройства назначения
0x58 (88)	BadSignature	Недопустимая подпись
0x59 (89)	BadSourceAddress	Недопустимый адрес источника
0x5A (90)	BadTimestamp	Недопустимая временная метка
0x5B (91)	CannotUseKey	Невозможно использовать ключ
0x5C (92)	CannotVerifyMessageID	Невозможно проверить ID сообщения
0x5D (93)	CorrectKeyRevision	Правильная ревизия ключа
0x5E (94)	DestinationDeviceIDRequired	Требуется ID устройства назначения
0x5F (95)	DuplicateMessage	Дублирующееся сообщение
0x60 (96)	EncryptionNotConfigured	Шифрование не настроено
0x61 (97)	EncryptionRequired	Требуется шифрование
0x62 (98)	IncorrectKey	Неверный ключ
0x63 (99)	InvalidKeyData	Недопустимые данные ключа
0x64 (100)	KeyUpdateInProgress	Идёт обновление ключа
0x65 (101)	MalformedMessage	Некорректное сообщение
0x66 (102)	NotKeyServer	Устройство не является сервером ключей
0x67 (103)	SecurityNotConfigured	Безопасность не настроена
0x68 (104)	SourceSecurityRequired	Требуется безопасность источника
0x69 (105)	TooManyKeys	Слишком много ключей
0x6A (106)	UnknownAuthenticationType	Неизвестный тип аутентификации
0x6B (107)	UnknownKey	Неизвестный ключ
0x6C (108)	UnknownKeyRevision	Неизвестная ревизия ключа
0x6D (109)	UnknownSourceMessage	Неизвестное исходное сообщение
0x6E (110)	NotRouterToDnet	Не маршрутизатор для Dnet
0x6F (111)	RouterBusy	Маршрутизатор занят
0x70 (112)	UnknownNetworkMessage	Неизвестное сетевое сообщение
0x71 (113)	MessageTooLong	Сообщение слишком длинное
0x72 (114)	SecurityError	Ошибка безопасности
0x73 (115)	AddressingError	Ошибка адресации
0x74 (116)	WriteBdtFailed	Ошибка записи в таблицу Broadcast Distribution Table (BDT)
0x75 (117)	ReadBdtFailed	Ошибка чтения BDT
0x76 (118)	RegisterForeignDeviceFailed	Ошибка регистрации внешнего устройства
0x77 (119)	ReadFdtFailed	Ошибка чтения Foreign Device Table (FDT)
0x78 (120)	DeleteFdtEntryFailed	Ошибка удаления записи из FDT
0x79 (121)	DistributeBroadcastFailed	Ошибка рассылки широковещательного сообщения

0x7A (122)	UnknownFileSize	Неизвестный размер файла
0x7B (123)	AbortApuTooLong	Прерывание: APDU слишком длинный
0x7C (124)	AbortApplicationExceededReplyTime	Прерывание: приложение превысило время ответа
0x7D (125)	AbortOutOfResources	Прерывание: недостаточно ресурсов
0x7E (126)	AbortTsmTimeout	Прерывание: таймаут Transaction State Machine (TSM)
0x7F (127)	AbortWindowSizeOutOfRange	Прерывание: размер окна вне допустимого диапазона
0x80 (128)	FileFull	Файл заполнен
0x81 (129)	InconsistentConfiguration	Несогласованная конфигурация
0x82 (130)	InconsistentObjectType	Несогласованный тип объекта
0x83 (131)	InternalError	Внутренняя ошибка устройства
0x84 (132)	NotConfigured	Не настроено
0x85 (133)	OutOfMemory	Недостаточно памяти
0x86 (134)	ValueTooLong	Значение слишком длинное
0x87 (135)	AbortInsufficientSecurity	Прерывание: недостаточный уровень безопасности
0x88 (136)	AbortSecurityError	Прерывание: ошибка безопасности

3.3 Описание правил дополнительного модуля валидации

ВАЖНО!

В данном разделе под `basnet`-объектом понимается совокупность `basnet`-параметров - `object_type:instance:property_id`.

Модуль валидации выполняет проверку входных данных файлов конфигурации. Помимо правил, описанных в пунктах выше, также проверяются и логические ошибки конфигурации, которые могут быть допущены по невнимательности (например, один и тот же сигнал опрашивается несколько раз). Соблюдение данных правил поможет избежать неожиданных ошибок и сделает поведение системы (система верхнего уровня + драйвер + `basnet`-устройства) более предсказуемым и надежным.

Ниже приведены основные правила, которые проверяются в этом модуле валидации:

- Проверка на уникальность выходных топиков в разделе `Read`, то есть для одного устройства не может быть несколько одинаковых топиков для разных `basnet`-объектов;
- Каждый `basnet`-объект в разделе `Read` должен присутствовать в этом разделе в единственном экземпляре, то есть нельзя читать один и тот же сигнал несколько раз, даже в разных периодах, даже если `parameter_name` у них отличаются;
- Каждый `basnet`-объект в разделе `Write` должен присутствовать в единственном экземпляре и в разделе `Read`, то есть управляемый объект обязательно должен регулярно опрашиваться;
- Возвращается предупреждение в случае, если для одного `basnet`-объекта, представленного в разделе `Write` и `Read` топик обратной связи отличается от выходного топика. Топик обратной связи — это топик, который формируется из ``/devices/${root_topic_mqtt}/controls/${parameter_name}``, описанных для сигнала в разделе `Write`. Управляющий топик — это топик, который формируется из ``/devices/${root_topic_mqtt}/controls/ ${parameter_name}{command_topic}``, описанных для сигнала в разделе `Write`. Зачем нужен топик обратной связи? При периодическом опросе сигнала, его состояние из устройства мы получаем с заданным периодом. А если управление пройдет между периодами, то обновленное значение, будет получено не сразу после успешного управления, а только после следующего опроса по периоду. Чтобы избежать этой ситуации, драйвер выполняет опрос сигнала, надо которым выполнялось управление сразу, не дожидаясь

периодического опроса и топик обратной связи - это как раз топик, с которым это значение выводится из драйвера в брокер.

- Топики обратной связи для сигналов в разделе Write должны быть уникальны между собой, в случае дубликатов драйвер запустится с предупреждением.
- Управляющие топики для сигналов в разделе Write должны быть уникальны между собой, относительно топиков на чтение и топиков обратной связи: т.е. драйвер не должен отправлять на брокер топика, на которые он же сам реагирует как на управляющие

ВАЖНО!

В случае, если нужна конфигурация, которая не выполняет принципы этих правил, то их соблюдение можно отключить ключом **--unsafe**. Проверка все равно будет выполнена, сообщения об ошибках все равно выпадут в терминал, но драйвер инициализируется и продолжит работу.

4. Примеры файлов конфигурации YAML

4.1 Bacnet.yaml

```
bacnet:
  enabled: true
  udp:
    # iface: lo
    port: 47811
    ip: 0.0.0.0/24
  # mqtt:
    # keepalive: 30
    # broker: localhost:1883

  # logger:
  #   write_in_logfile: false
  #   level_log: 3
  #   path_log: bacnet2mqtt.log
  #   max_size_log: 5
  #   max_backups_log: 70
  #   max_age_log: 30

devices:
  # bacnet_device1:
  #   file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device1.yaml
  # bacnet_device2:
  #   file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device2.yaml
  # bacnet_device3:
  #   file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device3.yaml
  # bacnet_device4:
  #   file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device4.yaml
  bacnet_device5:
    file: /home/isokolov-wsl/bacnet2mqtt/config/bacnet_device5.yaml
```

4.2 bacnet_device1.yaml

```
device:
  name: bacnet_device1
  enabled: true
  root_topic_mqtt: bacnet_dev1
  ip: 127.0.0.1:47809
data_model:
  read:
    - read_period_ms: 1000
    parameters:
      - parameter_name: param_AV
        object:
          object_type: AnalogValue
          instance: 1
          property_id: PresentValue
      - parameter_name: param_AI
        precision: 2
        transform: 0.1:0
        object:
          object_type: AnalogInput
          instance: 1
          property_id: PresentValue
      - parameter_name: param_AO
        object:
          object_type: AnalogOutput
          instance: 1
          property_id: PresentValue
      - parameter_name: param_BV
        object:
          object_type: BinaryValue
          instance: 1
          property_id: PresentValue
      - parameter_name: param_BI
        object:
          object_type: BinaryInput
          instance: 1
          property_id: PresentValue
      - parameter_name: param_BO
        object:
          object_type: BinaryOutput
          instance: 1
          property_id: PresentValue
      - parameter_name: param_CS
        object:
          object_type: CharacterString
          instance: 1
          property_id: PresentValue
  write:
    - parameter_name: param_AV
      command_topic: on
      object:
        object_type: AnalogValue
        instance: 1
    - parameter_name: param_AO
      command_topic: set
      object:
        object_type: AnalogOutput
        instance: 1
    - parameter_name: param_BV
      object:
        object_type: BinaryValue
        instance: 1
    - parameter_name: param_BO
      object:
        object_type: BinaryOutput
        instance: 1
    - parameter_name: param_CS
      object:
        object_type: CharacterString
        instance: 1
```

4.3 bacnet_device2.yaml

```
device:
  name: bacnet_device2
  enabled: true
  root_topic_mqtt: bacnet_dev2
  ip: 127.0.0.1:47810
  data_model:
    read:
      - read_period_ms: 1000
        parameters:
          - parameter_name: param_AV
            object:
              object_type: AnalogValue
              instance: 1
              property_id: PresentValue
          - parameter_name: param_AI
            precision: 2
            transform: 0.1:0
            object:
              object_type: AnalogInput
              instance: 1
              property_id: PresentValue
          - parameter_name: param_AO
            object:
              object_type: AnalogOutput
              instance: 1
              property_id: PresentValue
          - parameter_name: param_BV
            object:
              object_type: BinaryValue
              instance: 1
              property_id: PresentValue
          - parameter_name: param_BI
            object:
              object_type: BinaryInput
              instance: 1
              property_id: PresentValue
          - parameter_name: param_BO
            object:
              object_type: BinaryOutput
              instance: 1
              property_id: PresentValue
          - parameter_name: param_CS
            object:
              object_type: CharacterString
              instance: 1
              property_id: PresentValue
    write:
      - parameter_name: param_AV
        command_topic: on
        object:
          object_type: AnalogValue
          instance: 1
      - parameter_name: param_AO
        command_topic: set
        object:
          object_type: AnalogOutput
          instance: 1
      - parameter_name: param_BV
        object:
          object_type: BinaryValue
          instance: 1
      - parameter_name: param_BO
        object:
          object_type: BinaryOutput
          instance: 1
      - parameter_name: param_CS
        object:
          object_type: CharacterString
          instance: 1
```

5. BScan

Bscan — это CLI-утилита, предназначенная для работы с bacnet-устройствами, позволяющая в полуавтоматическом режиме собрать данные из bacnet-устройств и подготовить их для формирования конфигурационных файлов для драйвера bacnet2mqtt.

Данная утилита поддерживает две основные команды и флаги модификации для этих команд:

- `getinfo` - таргетный запрос информации об устройстве и формирование .xlsx отчета в формате для автоконфигуратора;
- `whois` - сканирование локальной сети на наличие bacnet-устройств.

Помимо флагов модификации команд, которые применяются только со своими командами, есть глобальные флаги, которые применяются ко всем командам, это:

- `-h, --help` вызов документации для конкретной команды;
- `-l, --level_log` Уровень логирования (1 - info, 2 - warn, 3 - debug, 4 - error) (default 1).

Данная информация отражена в команде `help` к самой утилите «./bscan help ## или `-h, --help`, но для команд только `-h` или `-help`».

Вызов будет следующим:

```
isokolov@ubuntu-pc:~/Documents/repos/bacnet2mqtt$ ./bscan help
bscan - это cli-утилита, предназначенная для работы с bacnet-устройствами

Usage:
  bscan [flags]
  bscan [command]

Available Commands:
  getinfo    Запрос списка объектов устройства
  help       Help about any command
  whois      Поиск устройств в локальной сети

Flags:
  -h, --help            help for bscan
  -l, --level_log int   Уровень логирования (1 - info, 2 - warn, 3 - debug, 4 - error) (default 1)
  -v, --version         Версия bscan

Use "bscan [command] --help" for more information about a command.
```

5.1 Команда `getinfo`

Команда `getinfo` предназначена для запроса и получения списка объектов ВАСnet-устройства по его адресу или идентификатору. Адресом здесь выступает ip-адрес и порт устройства, идентификатором - его `DeviceId`. Это минимальные исходные данные, необходимые для получения информации об устройстве.

Команда поддерживает следующие флаги:

Флаг	Описание
<code>-d, --DeviceID</code>	device id устройства
<code>--address</code>	IP:Port bacnet-устройства
<code>-h, --help</code>	help for getinfo
<code>-o, --output</code>	Файл для сохранения результата (по умолчанию вывод в консоль)

Команда работает следующим образом:

- Выполняется запрос списка объектов устройства;
- Выполняется запрос свойств присланных объектов;
- Результаты выводятся в консоль и, если указан выходной файл, записываются в `xlsx`-файл, в котором можно поправить необходимые поля и отправить в автоконфигуратор, который вернет `yaml`-файлы для драйвера `bacnet2mqtt`.

ВАЖНО!

Запрос свойств выполняется не для каждого присланного объекта, а только для тех объектов, которые поддерживаются драйвером (п.3.2.3, п.3.2.4).

Запрашиваются не все свойства, а только необходимые для получения представления о назначении этого `bacnet`-объекта. В основном это Имя объекта, Описание, Единицы измерения, Текущее значение. Для дискретных сигналов — это текст включенного/отключенного состояния, полярность. Для `MultiStateValue` — это список состояний и их описание и т.д. Вся полученная информация записывается в файл отчета.

Пример использования команды «./bscan getinfo --address 192.168.1.71:47808 -d 30 -o report.xlsx»:

```
isokolov@ubuntu-pc:~/Documents/repos/bacnet2mqtt$ ./bscan getinfo --address 192.168.1.71:47808 -d 30
2025-07-04T11:46:30.357 [INFO] Bacnet-адаптер успешно инициализирован.
2025-07-04T11:46:30.357 [INFO] [IP: 192.168.1.71:47808] Построение конфигурации из устройства
2025-07-04T11:46:30.994 [INFO] [IP: 192.168.1.71:47808 | DeviceID: 30] Конфигурация устройства:
1) Device:30
  - ModelName: 'Room_FC_2014'
  - VendorName: 'F. Chaxel, Thanks to Morten Kvistgaard, MIT licence, 2015'
  - ObjectName: 'RoomController.Simulator'
  - Description: 'Free RoomController Simulator, F. Chaxel 2015'
2) AnalogInput:0
  - Description: 'Indoor Temperature'
  - Units: '62'
  - PresentValue: '20.7'
  - ObjectName: 'Temperature.Indoor'
3) AnalogInput:1
  - ObjectName: 'Temperature.Water'
  - Description: 'Glycol Water Temperature'
  - Units: '62'
  - PresentValue: '37.3'
```

5.2 Команда whois

Команда `whois` предназначена для сканирования локальной сети на указанном интерфейсе на наличие `basnet`-устройств.

Команда получает на вход имя сетевого интерфейса, с которого рассылается `broadcast` сообщение «`whois`». Устройства, получившие это сообщение в ответ так же `broadcast`'ом отправляют сообщение «`iam`». Список ответивших устройств с их `ip`, `DeviceId` и еще некоторой метаданной выводится в консоль.

Команда поддерживает следующие флаги:

Флаг	Описание
<code>--allinfo</code>	Получить всю информацию об устройствах (по умолчанию только IP и ID устройства)
<code>-h, --help</code>	help for whois
<code>-i, --iface</code>	Имя сетевого интерфейса, через который рассылается <code>broadcast</code>
<code>-o, --output</code>	Файл для сохранения результата (по умолчанию вывод в консоль)
<code>-p, --port</code>	Порт, по которому рассылается <code>broadcast</code> (default 47808)

Для определения `broadcast`-адреса, `bscan` использует адрес и маску переданного сетевого интерфейса.

ВАЖНО!

Для сканирования устройств сразу на нескольких интерфейсах достаточно указать все необходимые интерфейсы через запятую без пробелов в аргументе `-i, --iface`.

Флаг `--allinfo` позволяет сразу после получения списка устройств в сети выполнить запрос информации по каждому устройству, как это делает команда `getinfo`.

Флаг `-o, --output` позволяет записать всю полученную информацию через флаг `--allinfo` в указанный `.xlsx-файл`, аналогично как в команде `getinfo.h`

Пример использования команды «`./bscan whois -i wlo1`»:

```
isokolov@ubuntu-pc:~/Documents/repos/bacnet2mqtt$ ./bscan whois -i wlo1
2025-07-04T11:51:19.002 [INFO] Ищем basnet-устройства в локальной сети
2025-07-04T11:51:19.002 [INFO] Basnet-адаптер успешно инициализирован.
2025-07-04T11:51:21.183 [INFO] Найденные устройства:
- Address: '192.168.1.77', Device ID: '260001', Vendor ID: '260'
- Address: '192.168.1.71', Device ID: '30', Vendor ID: '260'
```

«. /bscan whois -i wlo1 --allinfo -o "report.xlsx"»

Пример выходного файла *report.xlsx*, сформированного по результатам выполнения команды *whois* с флагами *allinfo* и указанием записи отчета в файл:

Страница Devices:

number	device_name	enabled	address	root_topic_mqtt	max_quantity	max_apdu	timeout_s
Номер	Имя устройства	Статус	IP-Port	Корневой топик	Максимальное количество объектов в запросе	Максимальный размер APDU	Таймаут ответа
1	RoomController.Simulator	true	192.168.1.71:63070	l.am.RoomController.Simulator		100	1476
2	SimpleServer	true	192.168.1.77:47808	l.am.SimpleServer		100	1476

Страница Read:

number	link_to_device	parameter_name	object_type	object_instance	property_id	description	units	read_period_ms	present_value	active_level	inactive_level	priority	separator
Номер	Ссылка на устройство или библиотеку Devices	Имя параметра	Тип объекта	Идентификатор объекта	Идентификатор свойства	Описание	Единицы измерения	Период чтения	Текущее значение	Активный уровень	Неактивный уровень	Приоритет	Символ разрыва
1	RoomController.Simulator	Temperature Indoor	AnalogOutput	0	PresentValue	Indoor Temperature	DegreesCelsius	3000 21	-val+	-val+	-val+	-val+	-val+
2	RoomController.Simulator	Temperature Water	AnalogOutput	1	PresentValue	Ground Water Temperature	DegreesCelsius	3000 36	-val+	-val+	-val+	-val+	-val+
3	RoomController.Simulator	Temperature Outdoor	AnalogOutput	2	PresentValue	Outdoor Temperature	DegreesCelsius	3000 12	-val+	-val+	-val+	-val+	-val+
4	RoomController.Simulator	setPoint_Value	AnalogOutput	0	PresentValue	Effective setpoint	DegreesCelsius	3000 21	-val+	-val+	-val+	-val+	-val+
5	RoomController.Simulator	setpoint_1	AnalogOutput	1	PresentValue	setpoint 1	DegreesCelsius	3000 21	-val+	-val+	-val+	-val+	-val+
6	RoomController.Simulator	setpoint_2	AnalogOutput	2	PresentValue	setpoint 2	DegreesCelsius	3000 19	-val+	-val+	-val+	-val+	-val+
7	RoomController.Simulator	setpoint_3	AnalogOutput	3	PresentValue	setpoint 3	DegreesCelsius	3000 17	-val+	-val+	-val+	-val+	-val+
8	RoomController.Simulator	TestSetpointCSV	CharacterString	1	PresentValue	CSV Setpoint Tests	unknown	3000 Comfort.Eco+Vacancy	-val+	-val+	-val+	-val+	-val+
9	RoomController.Simulator	State.Heater	BinaryValue	1	PresentValue	Heater Status	unknown	3000 0	On	Off	0	-val+	-val+
10	RoomController.Simulator	State.Chiller	BinaryValue	1	PresentValue	Chiller Status	unknown	3000 0	On	Off	0	-val+	-val+
11	RoomController.Simulator	State	MultiStateValue	0	PresentValue	Controller State Command	unknown	3000 2	-val+	-val+	-val+	-val+	-val+
12	RoomController.Simulator	State.VentilationLevel	MultiStateValue	1	PresentValue	Ventilation level	unknown	3000 1	-val+	-val+	-val+	-val+	-val+
13	SimpleServer	ANALOG INPUT 1	AnalogInput	1	PresentValue	Percent	unknown	3000 0	-val+	-val+	-val+	-val+	-val+
14	SimpleServer	ANALOG OUTPUT 1	AnalogOutput	1	PresentValue	Hz/dBts	unknown	3000 0	-val+	-val+	-val+	-val+	-val+
15	SimpleServer	ANALOG VALUE 1	AnalogValue	1	PresentValue	Percent	unknown	3000 0	-val+	-val+	-val+	-val+	-val+
16	SimpleServer	BINARY INPUT 1	BinaryInput	1	PresentValue	unknown	unknown	3000 0	Active	Inactive	0	-val+	-val+
17	SimpleServer	BINARY OUTPUT 1	BinaryOutput	1	PresentValue	unknown	unknown	3000 0	Active	Inactive	0	-val+	-val+
18	SimpleServer	BINARY VALUE 1	BinaryValue	1	PresentValue	unknown	unknown	3000 0	Active	Inactive	0	-val+	-val+
19	SimpleServer	CHARACTER STRING VALUE 1	CharacterString	1	PresentValue	unknown	unknown	3000 0	-val+	-val+	-val+	-val+	-val+
20	SimpleServer	MULTI-STATE INPUT 1	MultiStateInput	1	PresentValue	unknown	unknown	3000 1	-val+	-val+	-val+	-val+	-val+
21	SimpleServer	MULTI-STATE OUTPUT 1	MultiStateOutput	1	PresentValue	unknown	unknown	3000 1	-val+	-val+	-val+	-val+	-val+
22	SimpleServer	MULTI-STATE VALUE 1	MultiStateValue	1	PresentValue	unknown	unknown	3000 1	-val+	-val+	-val+	-val+	-val+

Страница Write:

number	link_to_device	parameter_name	command_type	object_type	object_instance	property_id	description	units	present_value	active_level	inactive_level	priority	separator
Номер	Ссылка на устройство или библиотеку Devices	Имя параметра	Тип команды	Тип объекта	Идентификатор объекта	Идентификатор свойства	Описание	Единицы измерения	Текущее значение	Активный уровень	Неактивный уровень	Приоритет	Символ разрыва
1	RoomController.Simulator	setPoint_Value	on	AnalogOutput	0	PresentValue	Effective setpoint	DegreesCelsius	21	-val+	-val+	-val+	-val+
2	RoomController.Simulator	setpoint_1	on	AnalogOutput	1	PresentValue	setpoint 1	DegreesCelsius	21	-val+	-val+	-val+	-val+
3	RoomController.Simulator	setpoint_2	on	AnalogOutput	2	PresentValue	setpoint 2	DegreesCelsius	19	-val+	-val+	-val+	-val+
4	RoomController.Simulator	setpoint_3	on	AnalogOutput	3	PresentValue	setpoint 3	DegreesCelsius	17	-val+	-val+	-val+	-val+
5	RoomController.Simulator	TestSetpointCSV	on	CharacterString	1	PresentValue	CSV Setpoint Tests	unknown	Comfort.Eco+Vacancy	-val+	-val+	-val+	-val+
6	RoomController.Simulator	State.Heater	on	BinaryValue	1	PresentValue	Heater Status	unknown	0	On	Off	0	-val+
7	RoomController.Simulator	State.Chiller	on	BinaryValue	1	PresentValue	Chiller Status	unknown	0	On	Off	0	-val+
8	RoomController.Simulator	State	on	MultiStateValue	0	PresentValue	Controller State Command	unknown	2	-val+	-val+	-val+	-val+
9	RoomController.Simulator	State.VentilationLevel	on	MultiStateValue	1	PresentValue	Ventilation level	unknown	1	-val+	-val+	-val+	-val+
10	SimpleServer	ANALOG OUTPUT 1	on	AnalogOutput	1	PresentValue	unknown	unknown	0	-val+	-val+	-val+	-val+
11	SimpleServer	ANALOG VALUE 1	on	AnalogValue	1	PresentValue	Percent	unknown	0	-val+	-val+	-val+	-val+
12	SimpleServer	BINARY OUTPUT 1	on	BinaryOutput	1	PresentValue	unknown	unknown	0	Active	Inactive	0	-val+
13	SimpleServer	BINARY VALUE 1	on	BinaryValue	1	PresentValue	unknown	unknown	0	Active	Inactive	0	-val+
14	SimpleServer	CHARACTER STRING VALUE 1	on	CharacterString	1	PresentValue	unknown	unknown	1	-val+	-val+	-val+	-val+
15	SimpleServer	MULTI-STATE OUTPUT 1	on	MultiStateOutput	1	PresentValue	unknown	unknown	1	-val+	-val+	-val+	-val+
16	SimpleServer	MULTI-STATE VALUE 1	on	MultiStateValue	1	PresentValue	unknown	unknown	1	-val+	-val+	-val+	-val+

ВАЖНО!

В нормальном режиме на стадии эксплуатации драйвер *basnet2mqtt* должен использоваться только для опроса свойств *PresentValue* *basnet*-объектов устройства, поэтому выходной файл формируется на опрос именно этого свойства, а вся сопутствующая описательная метainформация прикладывается в дополнительных столбцах, на основе которой определяется необходимость в этом сигнале, его правильное именование и все остальное, что необходимо для вывода сигнала в Андромеду. Подробнее про формат таблицы смотри в документации РП. Автоконфигуратор YAML.

Лист внесения изменений

Версия документа	Дата публикации	Авторы изменений	Описание изменений
rev_01	18.11.2025	Тимонин А.А.	Создан документ

