aprotech Kaspersky IoT company

Руководство пользователя

OPC UA Client & MQTT Publisher

Оглавление

1 О приложениях OPC UA Client и MQTT Publisher

Приложения OPC UA Client и MQTT Publisher (далее также *приложения*) представляют собой прикладное программное обеспечение, созданное для работы на платформе кибериммунной системы Kaspersky IoT Secure Gateway 1000 на базе операционной системы KasperskyOS.

Далее мы представим полное описание приложений и руководство по работе с ними. Информацию о Kaspersky IoT Secure Gateway 1000 вы можете узнать в документации к Kaspersky IoT Secure Gateway 1000.

Приложение OPC UA Client получает по протоколу OPC UA данные от сервера OPC UA, расположенного во внутренней сети предприятия. Приложение MQTT Publisher передает полученные данные по протоколу MQTT в MQTT-брокер с шифрованием TLS. Kaspersky IoT Secure Gateway 1000 обеспечивает безопасный сбор по OPC UA, конвертацию данных из протокола OPC UA в протокол MQTT и однонаправленную передачу данных от сервера OPC UA в MQTT-брокер.

Типовая схема развертывания Kaspersky IoT Secure Gateway 1000 в качестве типа устройства однонаправленный шлюз (диод данных) предполагает следующее:

1. Устройство представляет собой программный однонаправленный шлюз.
2. Сетевые стеки, относящиеся к внутренней и внешней сетям, разделены на уровне процессов.
3. Передача данных между внутренней и внешней сетями возможна только через специальный программный интерфейс MessageConsumer.

   Он обеспечивает однонаправленную передачу данных из внутренней сети к информационным системам во внешней сети. Для обеспечения конфиденциальности передаваемой информации используется протокол TLS.

   Программный интерфейс MessageConsumer реализован в следующих приложениях:

   - Приложение OPC UA Client для обработки трафика из внутренней сети.
   - Приложение MQTT Publisher для обработки трафика во внешней сети.
4. Приложение OPC UA Client подключено к внутренней сети.

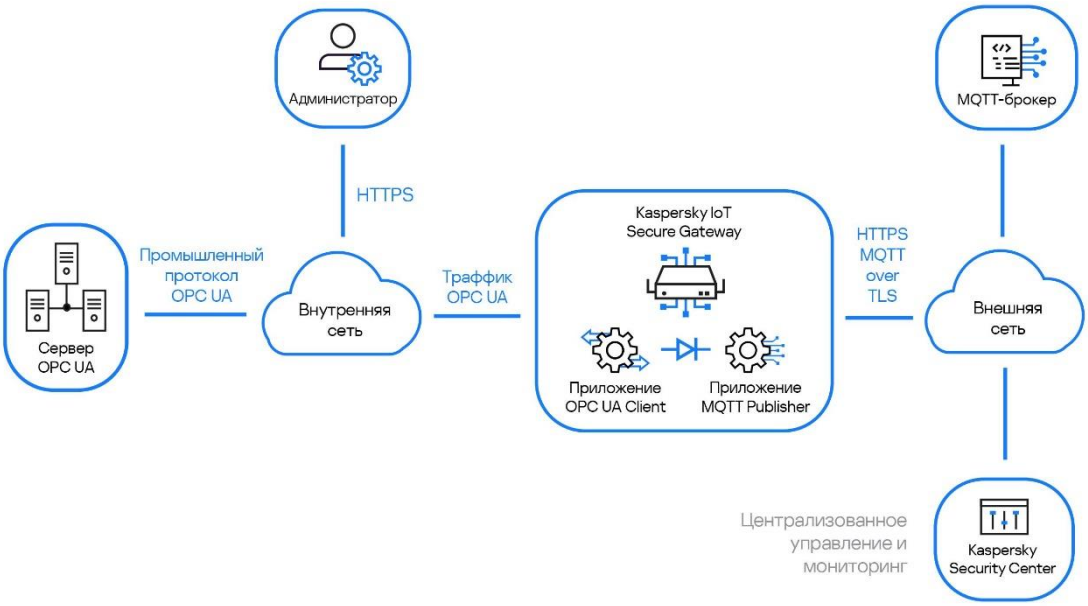

Рисунок 1. Типовая схема развёртывания Kaspersky IoT Secure Gateway 1000 с установленными приложениями OPC UA Client и MQTT Publisher.

Общая информация о схеме развёртывания Kaspersky IoT Secure Gateway 1000 представлена в документации к Kaspersky IoT Secure Gateway 1000.

Установку и предварительную настройку приложений на Kaspersky IoT Secure Gateway 1000 выполняют специалисты ООО «НПО АПРОТЕХ» или его партнеры.

1.1. Комплект поставки

В комплект поставки приложений входят следующие компоненты:

- Приложение OPC UA Client.
- Приложение MQTT Publisher.
- Файл с информацией о стороннем коде legal_notices.pdf.

1.2. Аппаратные и программные требования

1.2.1. Требования для работы приложений

Приложения работают только на Kaspersky IoT Secure Gateway 1000.

Необходимо настроить сервер OPC UA для приёма данных от оборудования и отправки данных в приложение OPC UA Client. Вы можете ознакомиться со спецификацией протокола OPC UA на сайте разработчика. Приложение поддерживает протокол OPC UA только версии 1.04.

Необходимо настроить MQTT-брокер для приема данных от приложения MQTT Publisher. Вы можете ознакомиться со спецификацией протокола MQTT на сайте разработчика. Приложение поддерживает протокол MQTT только версии 3.1.1.

1.2.2. Требования для настройки и диагностики приложений

Для настройки и диагностики приложений вам потребуется компьютер под управлением операционной системы Windows.

На компьютере должны быть установлены следующие прикладные программы:

- Программа для редактирования простого текста. Рекомендуется использовать текстовый редактор с поддержкой подсветки синтаксиса JSON.
- Браузер Google™ Chrome™ версии 118 и выше или Mozilla™ Firefox™ версии 118 и выше для доступа к веб-интерфейсу Kaspersky IoT Secure Gateway 1000

2 Что нового

В Kaspersky IoT Secure Gateway 1000 версии 3.0 реализованы следующие функции и возможности, значимые для работы приложений OPC UA Client и MQTT Publisher:

- Kaspersky IoT Secure Gateway 1000 действует в качестве программной платформы, которая поддерживает пограничные вычисления (англ. edge computing). Приложения располагаются на этой программной платформе, запускаются в изолированной среде и управляются с помощью платформы.
- Kaspersky IoT Secure Gateway 1000 работает в качестве однонаправленного шлюза (диода данных). Приложения OPC UA Client и MQTT Publisher запускаются, только когда Kaspersky IoT Secure Gateway 1000 работает в режиме однонаправленного шлюза. В документации к Kaspersky IoT Secure Gateway 1000 представлена информация о других режимах работы системы.
- Доступно управление приложениями с помощью веб-плагина для Kaspersky Security Center 14.2 Web Console. Включая такие действия как:
    - Скачивание и установка приложений; настройка конфигурации; запуск и остановка приложений, а также их удаление.
    - Работа с сертификатами приложений, включая добавление, обновление и удаление сертификатов приложений. Сертификат приложения — это специальный файл цифровой подписи, обеспечивающий безопасную работу приложения в Kaspersky IoT Secure Gateway 1000.
    - Настройка маршрутов передачи данных между приложениями, включая создание, изменение и удаление маршрута для приложения.
- Возможность ручного изменения конфигурации Kaspersky IoT Secure Gateway 1000 с помощью веб-интерфейса. В частности, таким образом возможно настроить перезапуск для приложений.
- Возможность выгружать журналы работы приложений с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000. Kaspersky IoT Secure Gateway 1000 записывает события, которые генерируются установленными приложениями, в журналы и обеспечивает сохранность этих журналов приложений при перезагрузке, выключении или обновлении системы.
- Возможность управлять уровнем журналирования приложений с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000. Доступен выбор между 6 уровнями логирования, которые различаются между собой по степени подробности и влиянию на производительность.

3 Работа с приложениями

Работе с приложениями предшествует настройка системы Kaspersky IoT Secure Gateway 1000. Перед тем как приступать к работе с приложениями, убедитесь, что вы выполнили шаги, предусмотренные документацией к Kaspersky IoT Secure Gateway 1000 для начала работы с системой. Подробнее об этом говорится в разделе «Предварительные условия работы с приложениями на Kaspersky IoT Secure Gateway 1000».

Приложениями можно управлять через веб-интерфейс или с помощью веб-плагина для Kaspersky Security Center 14.2 Web Console. Сценарии управления приложениями различаются в зависимости от того, используется ли веб-интерфейс или Kaspersky Security Center 14.2 Web Console. Далее мы рассмотрим оба сценария, ссылаясь на документацию к Kaspersky IoT Secure Gateway 1000.

Сценарий настройки передачи данных от сервера OPC UA в MQTT-брокер посредством приложений OPC UA Client и MQTT Publisher состоит из следующих этапов:

1. Настройка узлов передачи данных по протоколу OPC UA на сервере OPC UA. Вы можете ознакомиться со спецификацией протокола OPC UA на сайте разработчика.
2. Установка приложений OPC UA Client и MQTT Publisher на Kaspersky IoT Secure Gateway 1000.
3. Подготовка криптографического ключа и сертификатов для подключения по протоколу MQTT с шифрованием TLS.
    a. Подготовка криптографического ключа и файла, содержащего цепочку сертификатов, которыми производилась подпись сертификата MQTT-брокера.
    b. Подготовка криптографического ключа и файла, содержащего цепочку сертификатов, которыми производилась подпись сертификата приложения MQTT Publisher. Опционально, для клиентской аутентификации. Подробнее о работе с сертификатами читайте в разделе «Защита соединения по протоколу MQTT».
4. Настройка параметров приложений через веб-интерфейс или с помощью Kaspersky Security Center. Подробные инструкции, относящиеся к этому шагу, приведены в разделе «Настройка конфигурации приложений».
5. Настройка маршрутизации приложений для корректной передачи данных от приложения OPC UA Client приложению MQTT Publisher и как следствие этого от сервера OPC UA в MQTT-брокер.

3.1 Предварительные условия работы с приложениями на Kaspersky IoT Secure Gateway 1000

Перед началом работы с приложениями на Kaspersky IoT Secure Gateway 1000 необходимо выполнить ряд предварительных условий. Предварительные условия включают в себя:

- Установку и первоначальную настройку Kaspersky IoT Secure Gateway 1000.
- Подготовку сертификатов, необходимых для безопасной работы приложений в Kaspersky IoT Secure Gateway 1000, а также для поддержания зашифрованного канала связи передачи данных от приложения MQTT Publisher в MQTT-брокер.
- Подготовку дополнительного программного обеспечения, использующегося для работы с приложениями: Kaspersky Security Center 14.2 Web Console.

Предварительные условия представлены в списке ниже в рекомендованном порядке выполнения:

1) Подключить устройство Kraftway Рубеж-Н к сети и включить.

2) Подготовить устройство Kraftway Рубеж-Н к установке Kaspersky IoT Secure Gateway 1000.

3) Установить Kaspersky IoT Secure Gateway 1000, выбрав в качестве типа сетевого устройства однонаправленный шлюз. Шаги 2 и 3 выполняются специалистами ООО НПО "Апротех".

4) Создать и загрузить сертификаты администратора.

5) Настроить дату и время на Kaspersky IoT Secure Gateway 1000.

6) Настроить параметры сети на Kaspersky IoT Secure Gateway 1000.

7) Изменить сертификат веб-сервера на используемый в вашей организации.

8) Подключиться к веб-интерфейсу Kaspersky IoT Secure Gateway 1000. В процессе подключения в качестве администратора вам потребуется изменить учётные данные: имя пользователя и пароль. При изменении пароля форма ввода пароля будет указывать, что пароль соответствует требованиям, в том числе и в тех случаях, когда пароль в действительности не соответствует требованиям. Убедитесь самостоятельно в соответствии пароля требованиям, не ориентируясь на информацию от формы ввода пароля.

9) Установить веб-плагин для подготовки Kaspersky Security Center 14.2 Web Console к взаимодействию с Kaspersky IoT Secure Gateway 1000.

10) Добавить Kaspersky IoT Secure Gateway 1000 в управляемые устройства Kaspersky Security Center 14.2 Web Console.

11) Связать Kaspersky IoT Secure Gateway 1000 с Kaspersky Security Center 14.2 Web Console для последующего управления системой.

Чтобы работать с приложениями на Kaspersky IoT Secure Gateway 1000 вам потребуется учётная запись администратора. Для управления приложениями потребуется воспользоваться веб-интерфейсом Kaspersky IoT Secure Gateway 1000 или Kaspersky Security Center 14.2 Web Console. Возможности инструментов управления приложениями представлены в таблице ниже:

Возможности управления приложениями на Kaspersky IoT Secure Gateway 1000		
Функция	Веб-интерфейс Kaspersky IoT Secure Gateway 1000	Kaspersky Security Center 14.2 Web Console
Скачивание и установка приложений	Да	Да
Запуск и остановка приложений	См. раздел Известные ограничения	Да
Управление правилами запуска приложений	Да	Да
Удалений приложений	См. раздел Известные ограничения	Да
Настройка конфигурации приложений	Да	Да
Выгрузка журналов приложений	Да	Нет
Работа с сертификатами приложений	Нет	Да
Маршрутизация приложений	Да	Да

Помимо различий в наборе функций, следует учитывать, что для работы с веб-интерфейсом потребуется компьютер, который имеет доступ к Kaspersky IoT Secure Gateway 1000 через внутреннюю сеть. Тогда как управление посредством Kaspersky Security Center 14.2 Web Console может осуществляться удалённо.

3.2 Установка и удаление приложений

3.2.1 Скачивание и установка приложений

Чтобы скачать и установить приложения на Kaspersky IoT Secure Gateway 1000 с помощью веб-интерфейса, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 5 нажмите на кнопку Установить в столбце Действие напротив приложений OPC UA Client и MQTT Publisher.

Чтобы скачать и установить приложения на Kaspersky IoT Secure Gateway 1000 с помощью Kaspersky Security Center 14.2 Web Console, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 7 установите флажок у приложений OPC UA Client и MQTT Publisher и нажмите на кнопку Сохранить в нижней части страницы.

Выбранные приложения будут скачаны и установлены в Kaspersky IoT Secure Gateway 1000. После синхронизации Kaspersky IoT Secure Gateway 1000 и Kaspersky Security Center в таблице приложений для них отобразится статус Установлено. Информация об успешном или неуспешном скачивании и установке приложений сохраняется в журнал событий.

Установленные приложения не обновляются автоматически. Чтобы обновить приложение, вам нужно сначала удалить установленную версию приложения и затем установить его новую версию. Повторная установка версии приложения, снятой с публикации, невозможна.

3.2.2 Удаление приложений

Чтобы удалить приложение с Kaspersky IoT Secure Gateway 1000 с помощью веб-интерфейса, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 3 в строке приложения, которое вы хотите удалить, нажмите на значок корзины в столбце Удаление и подтвердите удаление в открывшемся окне.

Чтобы удалить приложение с Kaspersky IoT Secure Gateway 1000 с помощью Kaspersky Security Center 14.2 Web Console, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 7 установите флажок у приложения OPC UA Client или MQTT Publisher и нажмите на кнопку Сохранить в нижней части страницы.

3.3 Настройка конфигурации приложений

Разделы ниже содержат информацию о способах настройки конфигурации приложений OPC UA Client и MQTT Publisher.

Для настройки конфигурации приложений через веб-интерфейс вам потребуется информация о структуре конфигурации Kaspersky IoT Secure Gateway 1000. Конфигурация приложения содержится в строке configContent, которая находится в объекте APP_CONFIGURATION.

Обратите внимание, если приложение находится в состоянии Запущено, его необходимо остановить и запустить заново, чтобы новые параметры конфигурации были применены.

3.3.1 Настройка приложения OPC UA Client

3.3.1.1 Настройка соединения по протоколу OPC UA Client

Приложение OPC UA Client получает данные от сервера OPC UA, расположенного во внутренней сети организации, по протоколу OPC UA, описанному в спецификации OPC Unified Architecture (унифицированная архитектура OPC). Вы можете ознакомиться со спецификацией протокола OPC UA на сайте разработчика. Приложение поддерживает протокол OPC UA только версии 1.04.

3.3.1.2 Настройка приложения OPC UA Client через веб-интерфейс

Чтобы настроить получение данных по протоколу OPC UA:

1. Откройте веб-интерфейс Kaspersky IoT Secure Gateway 1000.
2. Откройте раздел «Параметры» и далее вкладку «Конфигурация».
3. Найдите в тексте, представленном на вкладке, блок ru.aprotech.opcuaclient.
4. Скопируйте закодированный в формате Base64 текст с параметрами приложения, который находится в строке configContent.
5. Декодируйте текст из формата Base64 в формат JSON (например, с помощью сайта https://www.base64decode.org/).
6. Скопируйте получившийся текст с параметрами приложения в отдельный файл для последующего редактирования.
7. Укажите параметры OPC UA и их значения, соблюдая синтаксис JSON.
8. Кодируйте заполненный текст настроек обратно из формата JSON в формат Base64 (например, с помощью сайта https://www.base64encode.org/). Перед этим рекомендуем убедиться в соблюдении синтаксиса JSON, поскольку веб-интерфейс Kaspersky IoT Secure Gateway 1000 не сообщит, если в конфигурации, закодированной в формате Base64, будут какие-либо ошибки. Если запустить приложение с ошибками в конфигурации, приложение будет остановлено. В журнале Kaspersky IoT Secure Gateway 1000 появится сообщение о том, что приложение завершило работу с ошибкой.
9. Скопируйте получившийся текст в строку configContent в блоке ru.aprotech.mqttpublisher во вкладке «Конфигурация».
10. Нажмите на кнопку «Сохранить».

Пример текста настроек приложения OPC UA Client в формате JSON:

```
{
  "name": "OPC UA Client Example",
  "description": "KISG Applications Development (Example)",
  "url": "opc.tcp://192.168.1.254:4840",
  "readingCycle": 1,
  "userCredentials": null,
  "heartbeat": {
    "name": "Heartbeat",
    "timeout": 3
  },
  "nodes": [
    {
      "name": "Boolean",
      "nodeId": "ns=2;s=Boolean"
    },
    {
      "name": "Opcstress1",
      "nodeId": "ns=2;s=Opcstress1"
    },
    {
      "name": "Opcstress2",
      "nodeId": "ns=2;s=Opcstress2"
    }
  ]
}
```

Пример кода 1. Пример настроек OPC UA Client

Для редактирования файлов в формате JSON мы рекомендуем использовать текстовый редактор с поддержкой подсветки синтаксиса JSON. Это позволит избежать возможных ошибок (например, непарных скобок).

3.3.1.3 Настройка приложения OPC UA Client через KSC Web Console

Чтобы настроить конфигурацию приложения OPC UA Client с помощью Kaspersky Security Center 14.2 Web Console, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 8 укажите для приложения необходимые параметры, опираясь на описание параметров приложения OPC UA Client. Параметры указываются в текстовом поле Application configuration, при этом важно соблюдать синтаксис формата JSON. Указав параметры, нажмите кнопку Сохранить сначала в нижней части панели настройки, затем в нижней части страницы установленных приложений.

```
{
  "name": "OPC UA Client Example",
  "description": "KISG Applications Development (Example)",
  "url": "opc.tcp://192.168.1.254:4840",
  "readingCycle": 1,
  "userCredentials": null,
  "heartbeat": {
    "name": "Heartbeat",
    "timeout": 3
  },
  "nodes": [
    {
      "name": "Boolean",
      "nodeId": "ns=2;s=Boolean"
    },
    {
      "name": "Opcstress1",
      "nodeId": "ns=2;s=Opcstress1"
    },
    {
      "name": "Opcstress2",
      "nodeId": "ns=2;s=Opcstress2"
    }
  ]
}
```

Пример кода 2. Пример настроек OPC UA Client

3.3.1.4 Описание параметров приложения OPC UA Client

Параметры, отмеченные как обязательные, следует явно указать. Прочие параметры настраивать необязательно. Для необязательных параметров, не включенных в конфигурацию, может использоваться значение по умолчанию, предусмотренное протоколом OPC UA.

Спецификация, определяющая протоколы и механизм передачи данных в промышленных сетях, а также взаимодействие устройств в них.

Параметры, используемые для настройки приложения OPC UA Client

Имя параметра	Обязательный параметр	Тип данных	Описание	Возможные значения и примечания
name	Да	string	Имя OPC UA Client, принимающего данные от сервера OPC UA.	<OPC UA client name>. Пример: "Kaspersky IoT Secure Gateway 1000 OPC UA Client".
description	Нет	string	Описание OPC UA Client, принимающего данные от сервера OPC UA.	<OPC UA client description>. Пример: "Collect data from CNC by Kaspersky IoT Secure Gateway 1000".
url	Да	string	Адрес сервера OPC UA.	<схема>://<хост>:<порт>. Пример: "opc.tcp://192.168.177.7:4840".

Имя параметра	Обязательный параметр	Тип данных	Описание	Возможные значения и примечания
				Порт 4840 используется по умолчанию.
readingCycle	Нет	int	Частота считывания данных приложением (в секундах).	1. Целое значение не меньше 0. 0 — специальное значение, которое устанавливает использование максимальной частоты, доступной клиенту и серверу.
userCredentials	Нет	object	Блок параметров с учетными данными OPC UA Client на сервере OPC UA.	Блок параметров {username, password} с учетными данными пользователя. null – указывается, если вы хотите разрешить анонимное подключение клиента OPC UA к серверу OPC UA. В этом случае указывать значения username и password не требуется.
username	Нет	string	Имя учетной записи пользователя для авторизации на сервере OPC UA.	"username".
password	Нет	string	Пароль учетной записи пользователя для авторизации на сервере OPC UA.	"password"
heartbeat	Нет	object	Блок параметров, содержащий настройки сигнала работоспособности Kaspersky IoT Secure Gateway 1000, генерируемый OPC UA Client.	Блок параметров {id, name, timeout}. null. Если вы не добавите параметр heartbeat или укажете значение null, сигналы работоспособности отправляться не будут.
name	Нет	string	Имя узла данных.	<heartbeat node name>. Пример: "Heartbeat".
timeout	Нет	int	Период в секундах между генерацией сигналов работоспособности.	60. Требуется указать целое значение не

Имя параметра	Обязательный параметр	Тип данных	Описание	Возможные значения и примечания
				меньше 0. Значение по умолчанию — 30.
nodes	Да	array	Блок параметров узлов данных.	Блок параметров {name, nodeId}. Заполняется для каждого узла данных.
name	Да	string	Имя точки подключения отправителя. Параметр используется в процессе настройки маршрутизации приложений.	<node name>. Пример: "Temperature". Значение каждого параметра name в блоках параметров nodes в конфигурации OPC UA Client должно быть уникальным.
nodeId	Да	string	Идентификатор узла данных.	<namespace>, <nodeID>.
ns	Да	string	Идентификатор пространства имен сервера OPC UA.	"namespace".
nodeId	Да	string	Идентификатор узла данных в пространстве имен сервера OPC UA.	<nodeID>. Возможны два типа идентификатора: s (string) — строковое значение идентификатора узла данных. Например, "nodeId": "ns=1;s=Variable temperature". i (numeric) — числовое значение идентификатора узла данных. Например, "nodeId": "ns=2;i=2045".

3.3.1.5 Особенности настройки параметров безопасности OPC UA

В текущей версии приложения OPC UA Client не реализована возможность безопасного подключения по протоколу OPC UA.

3.3.2 Настройка приложения MQTT Publisher

Приложение MQTT Publisher передает данные в MQTT-брокер по протоколу MQTT. Вы можете ознакомиться со спецификацией протокола MQTT на сайте разработчика. Приложение MQTT Publisher поддерживает протокол MQTT только версии 3.1.1.

3.3.2.1 Защита соединения по протоколу MQTT

Чтобы передавать данные с помощью приложения MQTT Publisher с шифрованием TLS, потребуется загрузить в Kaspersky IoT Secure Gateway 1000 следующие файлы. Перечень файлов представлен ниже с указанием опциональности некоторых вариантов:

1. Файл, содержащий цепочку сертификатов, которыми производилась цифровая подпись сертификата MQTT-брокера.
2. Файл, содержащий цепочку сертификатов, которыми производилась цифровая подпись клиентского сертификата приложения MQTT Publisher. Опционально, для клиентской аутентификации.
3. Файл закрытого криптографического ключа приложения MQTT Publisher. Опционально, для клиентской аутентификации.

Также потребуется загрузить файлы сертификатов на сервер MQTT-брокера:

1. Файл, содержащий цепочку сертификатов, которыми производилась цифровая подпись сертификата MQTT-брокера.
2. Файл закрытого криптографического ключа MQTT-брокера.
3. Файл, содержащий цепочку сертификатов, удостоверяющая клиентский сертификат MQTT Publisher. Опционально, для клиентской аутентификации.

Цепочка сертификатов может состоять из одного самоподписанного сертификата.

Подробнее о работе с сертификатами приложений читайте в документации к Kaspersky IoT Secure Gateway 1000. Сертификаты и криптографические ключи, используемые приложением MQTT Publisher, должны быть в формате CRT, CER, DER или PEM. Длина ключа сертификата приложения должна составлять не менее 2048 бит.

Обратите внимание, в случае отзыва сертификата MQTT-брокера, вам потребуется получить новый сертификат у администратора MQTT-брокера и заменить отозванный сертификат в MQTT-брокере. Если этого не сделать, Kaspersky IoT Secure Gateway 1000 будет доверять как отозванному сертификату, так и новому, пока не истечет срок действия отозванного сертификата. При этом возможна ситуация, когда соединение, установленное по безопасному каналу, в действительности не будет являться безопасным.

Каждый раз, когда перевыпускается сертификат MQTT-брокера, потребуется обновить в Kaspersky IoT Secure Gateway 1000 полную цепочку сертификатов, в которую входит листовой сертификат MQTT-брокера.

3.3.2.2 Настройка приложения MQTT Publisher через веб-интерфейс

Чтобы настроить отправку данных:

1. Откройте веб-интерфейс Kaspersky IoT Secure Gateway 1000.
2. Откройте раздел «Параметры» и далее вкладку «Конфигурация».
3. Найдите в тексте, представленном на вкладке, блок: ru.aprotech.mqttpublisher.
4. Скопируйте закодированный в формате Base64 текст с параметрами приложения, который находится в строке configContent.

5. Декодируйте текст из формата Base64 в формат JSON (например, с помощью сайта https://www.base64decode.org/).
6. Скопируйте получившийся текст с параметрами приложения в отдельный файл для последующего редактирования.
7. Укажите параметры MQTT и их значения, соблюдая синтаксис JSON.
8. Кодируйте заполненный текст настроек обратно из формата JSON в формат Base64 (например, с помощью сайта https://www.base64encode.org/). Перед этим рекомендуем убедиться в соблюдении синтаксиса JSON, поскольку веб-интерфейс Kaspersky IoT Secure Gateway 1000 не сообщит, если в конфигурации, закодированной в формате Base64, будут какие-либо ошибки. Если запустить приложение с ошибками в конфигурации, приложение будет остановлено. В журнале Kaspersky IoT Secure Gateway 1000 появится сообщение о том, что приложение завершило работу с ошибкой.
9. Скопируйте получившийся текст в строку configContent в блоке ru.aprotech.mqttpublisher во вкладке «Конфигурация».
10. Нажмите на кнопку «Сохранить».

Настройки приложения будут применены после перезагрузки Kaspersky IoT Secure Gateway 1000.

Пример текста настроек приложения MQTT Publisher в формате JSON:

```
{
  "name": "MQTT Publisher Example",
  "description": "KISG Applications Development (Example)",
  "clientId": "KisgApplicationsDevelopmentExample0",
  "serverUri": "mqtts://192.168.2.1:8883",
  "userCredentials": null,
  "lastWill": {
    "topicName": "LastWill",
    "message": "LastMessage"
  },
  "topics": [
    {
      "name": "Heartbeat",
      "topicName": "Heartbeat"
    },
    {
      "name": "Consumer 1",
      "topicName": "FirstConsumer"
    },
    {
      "name": "Consumer 2",
      "topicName": "SecondConsumer"
    }
  ]
}
```

Пример кода 3. Пример настроек MQTT Publisher

Для редактирования файлов в формате JSON мы рекомендуем использовать текстовый редактор с поддержкой подсветки синтаксиса JSON. Это позволит избежать возможных ошибок (например, непарных скобок).

3.3.2.3 Настройка приложения MQTT Publisher через KSC Web Console

Чтобы настроить конфигурацию приложения MQTT Publisher с помощью Kaspersky Security Center 14.2 Web Console, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 8 укажите для приложения необходимые параметры, опираясь на описание параметров приложения MQTT Publisher. Параметры указываются в текстовом поле Application configuration, при этом важно соблюдать синтаксис формата JSON. Указав

параметры, нажмите кнопку Сохранить сначала в нижней части панели настройки, затем в нижней части страницы установленных приложений.

```
{
  "name": "MQTT Publisher Example",
  "description": "KISG Applications Development (Example)",
  "clientId": "KisgApplicationsDevelopmentExample0",
  "serverUri": "mqtts://192.168.2.1:8883",
  "userCredentials": null,
  "lastWill": {
    "topicName": "LastWill",
    "message": "LastMessage"
  },
  "topics": [
    {
      "name": "Heartbeat",
      "topicName": "Heartbeat"
    },
    {
      "name": "Consumer 1",
      "topicName": "FirstConsumer"
    },
    {
      "name": "Consumer 2",
      "topicName": "SecondConsumer"
    }
  ]
}
```

Пример кода 4. Пример настроек MQTT Publisher

3.3.2.4 Описание параметров приложения MQTT Publisher

Параметры, отмеченные как обязательные, необходимо настроить. Прочие параметры настраивать необязательно. Для необязательных параметров, не включенных в конфигурационный файл, может использоваться значение по умолчанию, предусмотренное протоколом MQTT.

Параметры, используемые для настройки приложения MQTT Publisher

Имя параметра	Обязательный параметр	Тип данных	Описание	Возможные значения и примечания
name	Да	string	Имя MQTT Publisher, который будет отправлять данные в MQTT-брокер.	<MQTT Publisher name>. Пример: "Kaspersky IoT Secure Gateway 1000 MQTT Publisher".
description	Нет	string	Описание MQTT Publisher, который будет отправлять данные в MQTT-брокер.	<MQTT Publisher description>. Пример: "Transfer data to MQTT Broker by Kaspersky IoT Secure Gateway 1000".
clientId	Да	string	Уникальный идентификатор MQTT Publisher.	"1". Значение clientId должно быть уникальным среди

				всех подключенных к MQTT-брокеру клиентов.
serverUri	Да	string	Адрес сервера, к которому будет подключаться MQTT Publisher.	<схема>://<хост>:<порт>. Пример: "ssl://192.168.188.8:8883". ssl, tls, wss, mqtts — схемы обращения к ресурсу, предусмотренные архитектурой. 8883 — порт по умолчанию.
userCredentials	Да	object	Блок параметров, который отвечает за аутентификацию MQTT Publisher на сервере.	Блок параметров {username, password} с учетными данными пользователя. null — указывается, если вы хотите разрешить анонимное подключение клиента MQTT к MQTT-брокеру. В этом случае не требуется заполнять поля username и password.
username	Нет	string	Имя учетной записи пользователя для авторизации на сервере MQTT.	"username".
password	Нет	string	Пароль учетной записи пользователя для авторизации на сервере MQTT.	"password".
lastWill	Нет	object	Блок параметров для настройки сообщения, которое уведомляет о некорректном отключении клиента (LWT-сообщение).	Блок параметров {topicName, message}. Приложение может указать LWT-сообщение при первом подключении к MQTT-брокеру. MQTT-брокер хранит это сообщение до тех пор, пока не обнаружит

				некорректное отключение приложения, а при обнаружении — отправит LWT-сообщение всем клиентам, подписавшимся на получение такого сообщения. При корректном отключении приложения, MQTT-брокер не отправляет такое сообщение.
topicName	Нет	string	Название MQTT-топика, который определяет информационный канал, на котором публикуется LWT-сообщение.	<topicName>. Пример: "LastWill".
message	Нет	string	Содержание LWT-сообщения.	<message>. Пример: "LastMessage".
keepAlive	Нет	int	Интервал, в течение которого MQTT-брокер может не получать сообщения от MQTT Publisher и при этом не разрывать соединение.	800. Значение по умолчанию: 120. Возможные значения: 0–65535. Если значение keepAlive равно нулю, сервер не будет обязан отключать клиента на основании бездействия клиента. Сервер может отключить клиента, который, по его мнению, неактивен или не отвечает на запросы, в любое время, независимо от значения keepAlive, предоставленного клиентом.
qualityOfService	Нет	int	Параметр, определяющий гарантию доставки сообщений.	1. Соглашение между отправителем сообщения (издателем) и получателем сообщения

				(подписчиком), которое определяет гарантию доставки для конкретного сообщения. В спецификации MQTT определены три уровня qualityOfService: 0 — не более одного раза: клиент публикует сообщения, не проверяя факт доставки до брокера. Сообщения могут быть потеряны или продублированы. 1 — по крайней мере один раз: брокер подтверждает доставку. Сообщения могут дублироваться, но доставка гарантирована. 2 — ровно один раз: обеспечивается гарантированная доставка сообщения, при этом исключается возможное дублирование. Значение по умолчанию: 1.
topics	Да	array of objects	Массив из блоков параметров MQTT-топиков.	Массив блоков параметров [{name, topicName}]. Отдельный блок параметров в массиве заполняется для каждого MQTT-топика.
name	Да	string	Имя точки подключения получателя. Параметр используется в процессе настройки маршрутизации приложений.	<name>. Пример: "Temperature". Каждое значение параметра name в объектах topic конфигурации MQTT Publisher должно быть уникальным.

topicName	Да	string	Название MQTT-топика.	<topicName>. Пример: "Heartbeat". См. также: Особенности заполнения названий MQTT-топиков.

3.3.2.5 Особенности заполнения названий MQTT-топиков

При заполнении значений topicName учитывайте следующие особенности:

- В названиях MQTT-топиков нельзя использовать подстановочные знаки: # и +. Также не рекомендуется использовать в названиях MQTT-топиков знак $.
- Название MQTT-топика не может быть пустым (должно содержать хотя бы один символ).
- Названия MQTT-топиков чувствительны к регистру.
- Названия MQTT-топиков могут содержать символ пробела.
- MQTT-топики, различающимися только символом / в начале или в конце названия, являются разными MQTT-топиками.
- Допустимо название MQTT-топика, состоящее только из символа /.
- Название MQTT-топика не должно содержать нулевой символ (NUL).
- Названия MQTT-топиков представляют собой строки в кодировке UTF-8, они не должны быть объемом более 65535 байт.

3.4 Настройка маршрутизации приложений

Настройка маршрутизации приложений необходима для корректной передачи данных от приложения OPC UA Client приложению MQTT Publisher и как следствие этого от сервера OPC UA в MQTT-брокер. Перед тем как приступать к настройке маршрутизации приложений убедитесь, что вы корректно настроили конфигурацию приложений.

Чтобы создать новый маршрут с помощью Kaspersky Security Center 14.2 Web Console воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 7 вам потребуется:

1. В раскрывающемся списке Приложение-отправитель выбрать приложение OPC UA Client.
2. В раскрывающемся списке Точка подключения отправителя выбрать точку подключения, обозначенную параметром name в блоке параметров nodes конфигурации приложения OPC UA Client.
3. В раскрывающемся списке Приложение-получатель выбрать приложение MQTT Publisher.
4. В раскрывающемся списке Точка подключения получателя выбрать точку подключения, обозначенную параметром name в блоке параметров topics конфигурации приложения MQTT Publisher.
5. Нажать на кнопку Сохранить в нижней части панели.

Маршрут для приложений будет создан и отобразится в таблице. По умолчанию новый маршрут создается активным. Приложения применят созданные маршруты после перезапуска Kaspersky IoT Secure Gateway 1000.

Ключевое условие корректной маршрутизации приложений — верное сопоставление параметров name у узлов данных OPC UA и MQTT-топиков. Обратите внимание, имена точек подключения должны быть уникальны в рамках одного приложения, но могут совпадать между приложениями.

Действуя аналогичным образом, вы можете изменить ранее созданные маршруты приложений, следуя документации к Kaspersky IoT Secure Gateway 1000. Вы также можете удалять ранее созданные маршруты.

Также вы можете создавать маршруты с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000. Следуйте инструкции ниже:

1. Откройте веб-интерфейс Kaspersky IoT Secure Gateway 1000.
2. Откройте раздел "Параметры" и далее вкладку "Конфигурация".
3. Найдите в тексте, представленном на вкладке, блок, посвящённый маршрутизации: APPS_ROUTING (объект APPLICATIOS —> объект APPS_ROUTING).
4. В блоке applications в параметре endpoints перечислите все точки подключения.
    a. Для OPC UA Client — это параметры name в блоке параметров nodes, которые вы указывали в конфигурации приложения.
    b. Для MQTT Publisher — это параметры name в блоке параметров topics, которые вы указывали в конфигурации приложения.
5. В блоке параметров routes задайте параметры для всех маршрутов, которые необходимо создать.
    a. В блоке параметров destination в параметре application_id укажите ru.aprotech.mqttpublisher.
    b. В параметре endpoint — необходимую точку подключения, обозначенную параметром name в блоке параметров topics конфигурации приложения MQTT Publisher.
    c. В блоке параметров source в параметре application_id укажите ru.aprotech.opcuaclient.

d. В параметре endpoint — необходимую точку подключения, обозначенную параметром name в блоке параметров nodes конфигурации приложения OPC UA Client.

e. В параметре active укажите true.

6. Повторите шаг 5 для всех маршрутов, которые требуется создать. Пример заполненного блока параметров APPS_ROUTING представлен ниже.

7. Нажмите на кнопку Сохранить.

```
"APPS_ROUTING": {
     "applications": [
       {
         "application_id": "ru.aprotech.opcuaclient",
         "endpoints": [
           "Provider 1",
           "Provider 2"
         ],
         "name": "OPC UA Client",
         "subtype": "Input",
         "type": "Network protocol converter"
       },
       {
         "application_id": "ru.aprotech.mqttpublisher",
         "endpoints": [
           "Heartbeat",
           "Consumer 1",
           "Consumer 2"
         ],
         "name": "MQTT Publisher",
         "subtype": "Output",
         "type": "Network protocol converter"
       }
     ],
     "routes": [
       {
         "active": true,
         "destination": {
           "application_id": "ru.aprotech.mqttpublisher",
           "endpoint": "Consumer 1"
         },
         "source": {
           "application_id": "ru.aprotech.opcuaclient",
           "endpoint": "Provider 1"
         }
       },
       {
         "active": true,
         "destination": {
           "application_id": "ru.aprotech.mqttpublisher",
           "endpoint": "Consumer 2"
         },
         "source": {
           "application_id": "ru.aprotech.opcuaclient",
           "endpoint": "Provider 2"
         }
       }
     ]
   }
 },
```

Пример кода 5. Пример блока конфигурации APPS_ROUTING

Особенности настройки маршрутизации приложений в текущей версии Kaspersky IoT Secure Gateway 1000:

- После внесения каких-либо изменений в конфигурацию Kaspersky IoT Secure Gateway 1000 с помощью веб-интерфейса в параметре active у существующих маршрутов приложений автоматически проставляется значение false. Для корректной работы маршрутов потребуется заново указать для параметров active значение true.
- Не предусмотрена возможность деактивации маршрутов. Маршруты могут быть созданы, изменены или удалены. При этом у маршрутов не может иного значения в параметре active, кроме true.
- Если с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000 указать для маршрута в параметре active значение false, то при просмотре того же маршрута с помощью Kaspersky Security Center 14.2 Web Console у него будет указано значение «Активный». При этом Kaspersky Security Center 14.2 Web Console предложит «Сохранить изменения». При сохранении изменений в конфигурации, которая открывается в веб-интерфейсе, у маршрута в параметре active будет указано значение true.
- Для того, чтобы созданные или измененные маршруты начали работать, потребуется перезагрузить Kaspersky IoT Secure Gateway 1000.

3.5 Запуск и остановка приложений

Приложения должны быть в состоянии Запущено, чтобы они могли выполнять свои функции.

Чтобы запустить приложение с помощью веб-интерфейса, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 3 нажмите на кнопку Запустить в столбце Управление в строке приложения OPC UA Client или MQTT Publisher.

Условия запуска приложений:

- Приложение установлено и сконфигурировано без ошибок и находится в состоянии Остановлено.
- Для приложения выбрано правило запуска Запуск вручную или Автозапуск.

Чтобы остановить приложение с помощью веб-интерфейса, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 3 нажмите на кнопку Остановить в столбце Управление в строке приложения OPC UA Client или MQTT Publisher.

Вы можете остановить приложение, если оно сконфигурировано без ошибок и находится в состоянии Запущено. Например, вам потребуется остановить приложение, если возникла необходимость обновить его конфигурацию. После внесения и сохранения изменений в конфигурацию приложение может быть запущено снова.

Чтобы запустить или остановить приложения с помощью Kaspersky Security Center 14.2 Web Console, воспользуйтесь инструкцией в соответствующем разделе документации к Kaspersky IoT Secure Gateway 1000. Следуйте указаниям инструкции, на шаге 7 установите флажок около приложений OPC UA Client и MQTT Publisher, и нажмите Запустить/Остановить в верхней части таблицы.

3.5.1 Изменение правил запуска и настройка перезапуска приложений

Вы можете настроить, как приложение будет запускаться в Kaspersky IoT Secure Gateway 1000 (автоматически или вручную), или запретить запуск приложения с помощью веб-интерфейса или с помощью Kaspersky Security Center 14.2 Web Console. Если вы изменили правило запуска для запущенного приложения, правило запуска будет применено только после остановки приложений.

Чтобы настроить перезапуск приложений, потребуется произвести ручное изменение конфигурации Kaspersky IoT Secure Gateway 1000 с помощью веб-интерфейса. За перезапуск приложений отвечает ключ restart_on_failure в конфигурации Kaspersky IoT Secure Gateway 1000, который активирует режим перезапуска приложения при нештатном завершении работы.

3.6 Работа с журналами приложений

Разделы ниже содержат информацию о выгрузке журналов приложений с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000 и об управлении уровнями журналирования.

3.6.1 Выгрузка журналов приложений с помощью веб-интерфейса

Kaspersky IoT Secure Gateway 1000 записывает события, которые генерируются установленными приложениями, в журналы. Журналы приложений понадобятся вам для диагностики работы приложений и обращения в техническую поддержку.

Как выгрузить файлы журналов приложений с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000, указано в технической документации к Kaspersky IoT Secure Gateway 1000.

3.6.2 Управление уровнями журналирования

Kaspersky IoT Secure Gateway 1000 поддерживает шесть уровней журналирования. Уровни журналирования представлены в таблице ниже и ранжированы по степени подробности сведений, сохраняемых в журнале.

Уровень журналирования	Степень подробности	Описание	Пример
0	Critical	Фиксируются только сообщения о нештатных ситуациях, приводящих к аварийной остановке приложения.	[04-09-2023 13:48:19][Critical][ru.aprotech.jsonreceiver][3562:3563][][MessageRouterImpl.cpp:GetMessageConsumerConnectionInfo:147]* Message Router found 0 Message Consumers, but expected 1 exactly
1	Error	Фиксируются сообщение о нештатных ситуациях, прерывающей работу операции (например, передачу данных между приложениями).	[04-09-2023 13:48:19][Error][ru.aprotech.jsonreceiver][3562:3563][][MessageRouterImpl.cpp:GetMessageConsumerConnectionInfo:147]* Message Router found 0 Message Consumers, but expected 1 exactly
2	Warning	Фиксируются сообщение о нештатных ситуациях, не препятствующих работе операции.	[04-09-2023 13:48:18][Warning][ru.aprotech.mqttpublisher][3981:4200][][Socket.cpp:Connect:90]* Failed to connect to 192.168.2.1:8883
3	Info	Фиксируется информация о штатном выполнении операции.	[04-09-2023 13:48:19][Info][ru.aprotech.jsonreceiver][3562:3563][][MessageRouterImpl.cpp:DoUp:65]* Message Router will make next attempt after timeout
4	Debug	Фиксируется подробная техническая информация о выполнении операции.	[04-09-2023 13:48:19][Debug][ru.aprotech.jsonreceiver][3562:3563][][MessageRouterImpl.cpp:GetMessageConsumerConnectionInfo:147]* Message Router found 0 Message Consumers, but expected 1 exactly
5	Trace	Фиксируется максимально возможный объём информации, используемый для наиболее детальной отладки. При включении может значительно влиять на производительность.	[04-09-2023 13:49:40][Trace][ru.aprotech.jsonreceiver][3562:4345][][Client.cpp:OnNewLine:116] CRT {"source": {"name": "JSON Receiver example", "port": "Generator"}, "dataItem": {"timestamp": "2023-09-04T14:29:31.503Z", "timestampSource": null, "value": "123", "status": "00000000"}}

Таким образом, в зависимости от имеющейся потребности в информации о работе приложений и задач по диагностике их состояния, стоит устанавливать различные уровни журналирования. По умолчанию для всех приложений используется уровень журналирования 4 (Debug). Если вам потребовалось, чтобы журнал содержал трассировку

элементов данных, установите уровень журналирования 5 (Trace). Уровень журналирования устанавливается для каждого приложения отдельно.

Чтобы установить требуемый уровень журналирования для приложения, воспользуйтесь следующей инструкцией:

1. Откройте веб-интерфейс Kaspersky IoT Secure Gateway 1000.
2. Откройте раздел «Параметры» и далее вкладку «Конфигурация».
3. Найдите в тексте, представленном на вкладке, строку logLevel (объект APPLICATIOS —> список объектов applications —> ключ logLevel).
4. Установите требуемый уровень логирования из списка выше. Например, "logLevel":"Warning"
5. Нажмите на кнопку «Сохранить».

4 Диагностика и обращение в техническую поддержку

Если вам не удается настроить приложения и вы не нашли решения вашего вопроса в документации или вы столкнулись с какими-либо неполадками в работе приложений, а также если вам потребовалось переустановить Kaspersky IoT Secure Gateway 1000 на устройство Kraftway Рубеж-Н, обратитесь в службу технической поддержки ООО «НПО АПРОТЕХ» по электронной почте: support@aprotech.ru. К обращению нужно приложить:

- Подробное описание проблемы.
- Настройки приложений OPC UA Client и MQTT Publisher, а также настройки сервера OPC UA и MQTT-брокера.
- Файлы с журналами приложений.
- Название организации и контактные данные для обратной связи.

5 Лицензирование

Условия использования приложений изложены в Лицензионном договоре или подобном документе, на основании которого используется программа.

6 Предоставление данных

Приложения OPC UA Client и MQTT Publisher не собирают, не используют и не обрабатывают пользовательские персональные данные.

7 Известные ограничения

7.1 Общие ограничения

Следующие ограничения затрагивают оба приложения, OPC UA Client и MQTT Publisher, а также программную платформу:

- Если с помощью веб-интерфейса Kaspersky IoT Secure Gateway 1000 удалить с устройства какое-либо одно из приложений: OPC UA Client или MQTT Publisher, то второе приложение будет автоматически тоже удалено.
- Kaspersky IoT Secure Gateway 1000 поддерживает передачу данных приложениями по не более чем 256 маршрутам одновременно.
- При запуске или остановке приложений вручную запускать или останавливать приложения OPC UA Client и MQTT Publisher необходимо только парой. Для запуска необходимо сначала запустить MQTT Publisher, затем OPC UA Client. Останавливать в обратном порядке: сначала OPC UA Client, затем MQTT Publisher. Рекомендуемая конфигурация запуска приложений — автоматический запуск для обоих приложений.
- Во время подключения к веб-интерфейсу Kaspersky IoT Secure Gateway 1000 при изменении пароля форма ввода пароля будет указывать, что пароль соответствует требованиям, в том числе и в тех случаях, когда пароль в действительности не соответствует требованиям. Убедитесь самостоятельно в соответствии пароля требованиям, не ориентируясь на информацию от формы ввода пароля.
- В случае ошибки передачи данных между приложениями OPC UA Client и MQTT Publisher нет достоверного способа понять, какие данные были доставлены корректно и какие нет. Возможны ситуации, когда некоторые данные будут отмечены в журнале как утерянные, даже если в действительности они были переданы корректно.
- Размер пространства для хранения журналов приложений OPC UA Client и MQTT Publisher имеет ограничение по 50 мб для каждого из приложений.
- Kaspersky IoT Secure Gateway 1000 не оборудован встроенным источником бесперебойного питания, поэтому рекомендуем использование внешнего ИБП во избежание потери данных в случае нереднамеренного отключения питания.
- При изменении конфигурации любого из приложений (OPC UA Client или MQTT Publisher) маршруты передачи данных инвалидируются. В такой ситуации Kaspersky Security Center 14.2 Web Console автоматически переведёт все маршруты в состояние «Активен» и предложит «Сохранить изменения».
- Инвалидация (перевод маршрутов из состояния «Активен») носит уведомительный характер. Kaspersky IoT Secure Gateway 1000 оповещает приложения об инвалидации маршрутов, но не запрещает передачу данных по ним.
- В Kaspersky Security Center 14.2 Web Console не предусмотрена возможность скачивания файлов, загруженных пользователем на страницу с конфигурацией приложений (например, файлов сертификатов).
- Kaspersky Security Center 14.2 Web Console при попытке установить, удалить или обновить приложение на управляемом с его помощью Kaspersky IoT Secure Gateway 1000 в меню «Программы» во вкладке «Параметры программы» в подменю «Менеджер приложений» не отображает список доступных приложений.

7.2 Ограничения OPC UA Client

Приложение OPC UA Client имеет следующие ограничения поддержки протокола OPC UA:

- Отсутствует возможность безопасного подключения по протоколу OPC UA. Подключение выполняется при использовании политики безопасности «None». Аутентификация на сервер OPC UA производится по имени пользователя и паролю. Учётные данные передаются в открытом виде. Также возможно анонимное подключение посредством указания параметра null в блоке параметров userCredentials.
- Поддерживаются только следующие типы данных, описанные в спецификации OPC UA:
    - Boolean;
    - SByte;
    - Byte;
    - Int16;
    - UInt16;
    - Int32;
    - UInt32;
    - Int64;
    - UInt64;
    - Float;
    - Double;
    - String;
    - DateTime;
    - XmlElement;
    - NodeId (только numeric и string);
    - ExpandedNodeId (только numeric и string);
    - StatusCode;
    - QualifiedName;
    - LocalizedText (частично);
    - Variant.
- Полученные по протоколу OPC UA данные типа Double и Float, округляются с точностью до шести значащих цифр.
- Для передачи данных по OPC UA сервер должен поддерживать наборы служб MonitoredItem и Subscription.
- Доступно подключение только одного клиента OPC UA к одному серверу OPC UA.

7.3 Ограничения MQTT Publisher

Приложение MQTT Publisher имеет следующие ограничения поддержки протокола MQTT:

- Доступно подключение только одного клиента MQTT к одному MQTT-брокеру.
- MQTT Publisher использует значение «1» для флага Clean Session при каждом подключении к MQTT-брокеру.
- Значение параметра qualityOfService является общим для всех публикуемых сообщений от MQTT Publisher в топики (параметр topics), включая служебные топики (heartbeat, lastWill).
- Значение параметра qualityOfService не может быть настроено для каждого публикуемого сообщения от MQTT Publisher в топики (параметр topics).
- Клиент MQTT не использует флаг `retain` при отправке сообщений, а также для LWT-сообщения (сообщения, которое уведомляет о некорректном отключении клиента).

- Установка значения `0` для параметра `keepAlive` клиента MQTT не приводит к отключению механизма "keep alive" (механизма для отключения клиента на основании его бездействия).
- Клиент MQTT игнорирует отсутствие ответа от MQTT-брокера в течение длительного времени и не закрывает соединение.
- В случае обрыва соединения не более 10 публикуемых сообщений может быть утеряно после восстановления соединения и при наличии свободного места в буфере.
- Приложение MQTT Publisher может перестать передавать данные в MQTT-брокер после остановки и повторного запуска. Для восстановления корректной работы приложения потребуется перезагрузить Kaspersky IoT Secure Gateway 1000.
- Если установить конфигурацию, при которой для приложения MQTT Publisher будет выбран запуск вручную, а для приложения OPC UA Client автоматический запуск, то при включении Kaspersky IoT Secure Gateway 1000 оба приложения будут в состоянии остановлено. Обратная конфигурация (запуск вручную для приложения OPC UA Client, автоматический запуск — для MQTT Publisher) работает корректно.

7.4 Ограничения TLS

Приложение MQTT Publisher имеет следующие ограничения поддержки протокола TLS:

- Компонент Kaspersky IoT Secure Gateway 1000, отвечающий за поддержание зашифрованного канала связи передачи данных, не поддерживает использование поля subjectAltName и не позволяет установить соединение с MQTT-брокером, если поле subjectAltName использовано в сертификате.
- Компонент Kaspersky IoT Secure Gateway 1000, отвечающий за поддержание зашифрованного канала связи передачи данных требует, чтобы поле Common name в сертификате содержало IP-адрес MQTT-брокера.
- Поддерживаются версии протокола TLS не ниже 1.2.
- Поддерживаются только наборы шифров TLS:
  - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384;
  - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256;
  - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256;
  - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384;
  - TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256;
  - TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256;
  - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384;
- Поддерживаются только следующие алгоритмы цифровой подписи:
  - ecdsa_secp521r1_sha512;
  - ecdsa_secp384r1_sha384;
  - ecdsa_secp256r1_sha256;
  - ed25519;
  - ed448;
  - rsa_pss_pss_sha512;
  - rsa_pss_rsae_sha512;
  - rsa_pss_pss_sha384;
  - rsa_pss_rsae_sha384;
  - rsa_pss_pss_sha256;
  - rsa_pss_rsae_sha256;
  - rsa_pkcs1_sha384;
  - rsa_pkcs1_sha512;
  - rsa_pkcs1_sha256.

8 Другие источники информации

Дополнительные документы, к которым вы можете обратиться в процессе установки, настройки и использования приложений:

- Техническая документация Kaspersky IoT Secure Gateway 1000.
- Спецификация протокола OPC UA.
- Спецификация протокола MQTT.

9 Глоссарий

Kaspersky IoT Secure Gateway 1000

Кибериммунная система на базе операционной системы KasperskyOS с предварительно настроенным набором прикладного программного обеспечения. Kaspersky IoT Secure Gateway 1000 устанавливается на встраиваемый компьютер модели Kraftway Рубеж-H и предназначена для работы в качестве безопасного шлюза Интернета вещей (Internet of Things) в сети организации.

Kaspersky Security Center

Программа, предназначенная для централизованного решения основных задач по управлению и обслуживанию системы защиты сети организации. Программа предоставляет администратору доступ к детальной информации об уровне безопасности сети организации и позволяет настраивать все компоненты защиты, построенной на основе программ "Лаборатории Касперского".

Kaspersky Security Center 14.2 Web Console

Приложение (веб-приложение), предназначенное для контроля состояния системы безопасности сетей организации, находящихся под защитой приложений "Лаборатории Касперского".

KasperskyOS

Микроядерная операционная система для построения безопасных решений.

Message Queuing Telemetry Transport (MQTT)

Сетевой протокол, работающий поверх стека протоколов TCP/IP, предназначенный для обмена сообщениями между устройствами в Интернете вещей.

MQTT-брокер

Сервер, принимающий, фильтрующий и пересылающий сообщения по протоколу MQTT.

MQTT-топик

Иерархический путь к источнику данных, на базе которого отправляются сообщения по протоколу MQTT.

Open Platform Communications Unified Architecture (OPC UA)

Спецификация, определяющая протоколы и механизм передачи данных в промышленных сетях, а также взаимодействие устройств в них.

TLS

Безопасный протокол передачи данных в локальных сетях и в интернете с использованием шифрования. TLS используется для создания защищенных соединений между клиентом и сервером.

Безопасный шлюз Интернета вещей

Система, которая обеспечивает безопасную передачу пользовательского трафика между датчиками и платформой Интернета вещей.

Веб-интерфейс Kaspersky IoT Secure Gateway 1000

Инструмент для работы с Kaspersky IoT Secure Gateway 1000. Для подключения к веб-интерфейсу потребуется браузер, установленный на компьютере, который имеет доступ к Kaspersky IoT Secure Gateway 1000 через внутреннюю сеть.

Интернет вещей

Вычислительная сеть электронных устройств ("вещей"), оснащенных встроенными возможностями взаимодействия с внешней средой или друг с другом без участия человека.

Источник данных

Обособленный источник данных для обмена сообщениями между устройствами в Интернете вещей. Например, сервер OPC UA на управляющем контроллере станка.

Кибериммунная информационная система

Система, гарантирующая достижение целей безопасности во всех возможных сценариях использования системы, предусмотренных разработчиками.

Клиент

Участник клиент-серверного взаимодействия, делающий запросы к серверу и получающий на них ответы.

Корневой сертификат

Сертификат корневого удостоверяющего центра.

Корневой удостоверяющий центр

Удостоверяющий центр, над которым нет вышестоящего удостоверяющего центра.

Криптографический ключ

Компонент пары криптографических ключей, используемых для асимметричной криптографии. Ключи могут быть открытыми или закрытыми.

Набор шифров

Совокупность шифров, работающих вместе и выполняющих различные криптографические функции, такие как генерация ключей и аутентификация. Наборы шифров описывают шаги, которые ключи должны выполнить, и порядок, в котором эти шаги выполняются.

Однонаправленный шлюз (диод данных)

Шлюз данных, который создан с помощью программных средств и который допускает передачу данных только в одну сторону. Представляет собой эффективное средство защиты от утечек конфиденциальной информации.

Программная платформа

Совокупность программного обеспечения и инструментов, предоставляемых разработчикам для создания и запуска приложений. Kaspersky IoT Secure Gateway 1000 выступает в качестве программной платформы для приложений OPC UA Client и MQTT Publisher.

Сервер

Участник клиент-серверного взаимодействия, выполняющий обработку запросов от клиента.

Сертификат конечного субъекта

Сертификат, содержащий в себе открытый криптографический ключ, который может быть использован для проверки или валидации конечного субъекта (например, клиента MQTT).

Сертификат

Структура данных с цифровой подписью, содержащая открытый криптографический ключ и идентификатор клиента или сервера.

Сертификат администратора

Сертификат, на основании которого осуществляется аутентификация пользователя в веб-интерфейсе Kaspersky IoT Secure Gateway 1000.

Событие

Запись, содержащая информацию об обнаружении данных в системе или во внутренней сети, которые требуют внимания сотрудника, ответственного за информационную безопасность в вашей организации, сохраняемая в памяти встраиваемого компьютера Kraftway Рубеж-Н.

Узел данных

Структурный элемент информационной модели OPC UA, содержащий данные и метаданные.

Уровень логирования

Режим работы журнала Kaspersky IoT Secure Gateway 1000, который определяет, информация о каких событиях фиксируется в журнале работы приложений, а также степень подробности этой информации.

Цепочка сертификатов

Объединение промежуточных сертификатов, в котором на пути от сертификата конечного субъекта до корневого сертификата может быть любое количество промежуточных сертификатов.

Цифровая подпись

Значение, вычисляемое с помощью криптографического алгоритма и добавляемое к данным таким образом, что любой получатель данных может использовать подпись для проверки происхождения и целостности данных.

Шифрование

Преобразование данных из читаемого формата в кодированный. Зашифрованные данные могут быть прочитаны или обработаны только после расшифровки.

10 Информация о стороннем коде

Информация о стороннем коде содержится в файле legal_notices.pdf.

11 Уведомления о товарных знаках

Зарегистрированные товарные знаки и знаки обслуживания являются собственностью их правообладателей.

Windows является товарным знаком группы компаний Microsoft.

Google и Google Chrome — товарные знаки Google LLC.

Mozilla и Firefox являются товарными знаками Mozilla Foundation в США и других странах.

Kraftway — зарегистрированный товарный знак ЗАО «Крафтвэй корпорэйшн ПЛС».