ТЕХНИЧЕСКИЙ ОБЗОР

Интеграция устройств с MQTT

Март 2022

AXIS COMMUNICATIONS

Краткая информация

Стандартный протокол обмена сообщениями MQTT способствует эффективному и надежному обмену данными между устройствами IoT и облачными приложениями. Он позволяет устройствам (через их клиенты MQTT) публиковать сообщения на общем брокере (сервере) MQTT, обеспечивающем связь с другими устройствами. Брокер следит за тем, кто что публикует и кто хочет просматривать данные, пересылая сообщения только тем клиентам, которые подписаны на соответствующую тему.

В типовой экосистеме VMS уведомления о событиях Axis, поступающие от устройств, традиционно транслируются в единую точку назначения через интерфейс API VAPIX/ONVIF с использованием протокола потоковой передачи RTSP. Но те же уведомления можно распространять с использованием протокола MQTT через встроенный клиент MQTT устройства (относится к устройствам, работающим под управлением AXIS OS 9.80 и более поздних версий). Это возможно как внутри экосистем VMS, так и за их пределами, и особенно эффективно при реализации через Интернет. Использовать и обрабатывать уведомления о событиях, публикуемые устройством Axis, могут несколько находящихся в сети подписанных клиентов MQTT. Также существуют аналитические приложения ACAP от Axis и сторонних поставщиков, в которых имеются собственные клиенты MQTT, разработанные для определенных систем, вариантов использования и подписчиков.

В качестве примера варианта использования, связанного с продукцией Axis, можно привести устройства для подсчета посетителей, которые могут отправлять по MQTT статистику в программное обеспечение для визуализации данных в облаке. Еще одним примером является датчик двери от стороннего производителя, поддерживающий по MQTT связь с сигнальным устройством, которое подает сигнал тревоги, и камерой, которая начинает съемку, каждый раз, когда открывается дверь.

Содержание

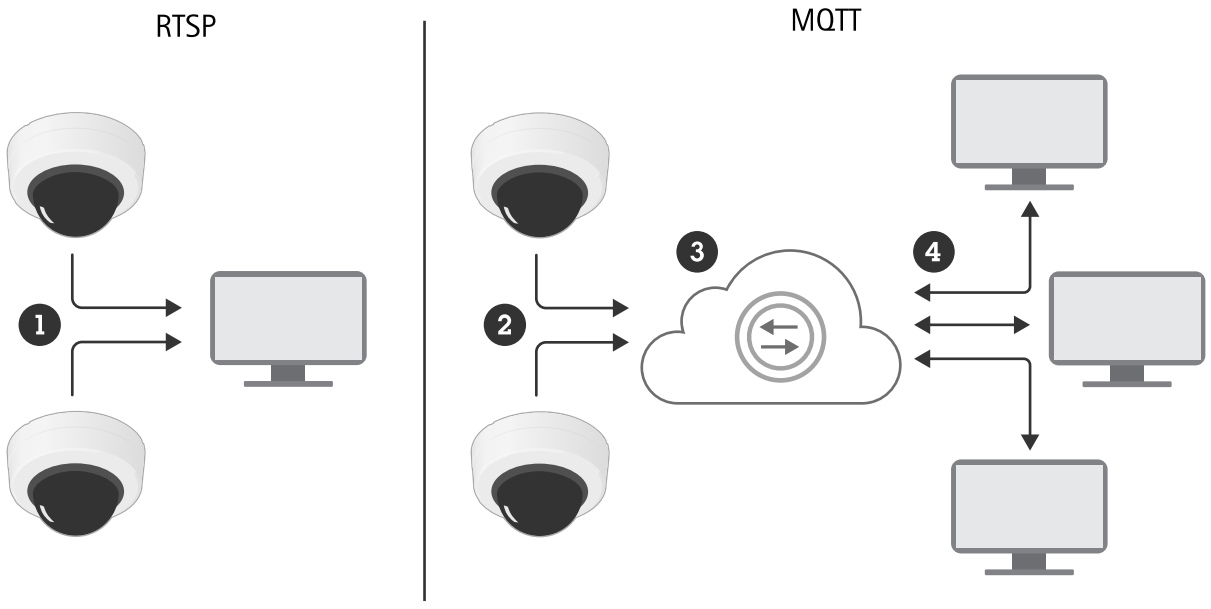
1 Введение

MQTT (Message Queuing Telemetry Transport) — это стандартный протокол обмена сообщениями для Интернета вещей (IoT). Он был разработан с целью упростить интеграцию IoT и используется в самых разных отраслях для подключения удаленных устройств с небольшим объемом кода и минимальными требованиями к пропускной способности сети. Клиент MQTT в AXIS OS позволяет упростить интеграцию данных и событий устройства в другие системы, которые не являются системами управления видео (VMS).

В настоящем техническом обзоре представлены технические особенности протокола MQTT, а также его преимущества, ограничения и типовые сценарии использования. В документе также содержатся сведения о клиентах MQTT в устройствах Axis и аналитических приложениях ACAP.

2 Протокол MQTT

Протокол MQTT основан на принципе «публикация — подписка». Это значит, что его алгоритм работы с сообщениями отличается от такового у протоколов RTSP и HTTP, основанных на принципе «запрос — отклик». В случае с RTSP одна сторона делает запрос, а другая сторона откликается. Вместо этого многие мобильные приложения для обмена сообщениями основаны на MQTT или аналогичных концепциях на основе принципа «публикация — подписка». Также существуют основанные на том же принципе протоколы, оптимизированные для закрытых и специализированных систем.



Протокол RTSP позволяет устанавливать связь по схеме «один — одному», тогда как MQTT также делает возможным связь «один — многим» и «многие — многим» через брокера.

1 *Поток событий*
2 *Публикация*
3 *Брокер MQTT*
4 *Подписка*

Концепция MQTT заключается в том, что все клиенты подключаются к общему брокеру MQTT (серверу), который следит за тем, кто что публикует и кто хочет просмотреть данные. Соединение

обычно представляет собой сеанс TCP на 1883 порте. Клиент также может подключаться через TLS (обычно 8883 порт) или с использованием WebSocket (обычно 1884/8884 порт).

Клиенты публикуют сообщения с указанием темы. Другой клиент может подписаться на ту или иную тему или с помощью подстановочных знаков получить все подтемы. В сообщении также содержится полезная нагрузка, обычно представляющая собой структуру данных JSON, строку или даже короткие двоичные данные. Публикующий не знает, подписываются ли другие клиенты. Брокер лишь пересылает сообщения клиентам, подписанным на темы.

В случае с MQTT ситуация несколько похожа на отправку статьи в журнал. Прочесть статью могут подписчики журнала, и это возможно в формате связи по принципу «один — одному» или «один — многим» (а MQTT позволяет устанавливать еще и связь «многие — многим»). Кроме того, статью можно читать еще долго после того, как она была впервые опубликована.

Для сравнения, использование протокола RTSP больше похоже на звонок по телефону. У команд всегда имеется одна исходная и одна целевая точка, связь всегда осуществляется по принципу «один — одному». Если в целевой точке на звонок не ответили, они пропустили сообщение.

Когда для распространения уведомлений о событиях Axis из устройств используется протокол MQTT, использовать и обрабатывать эти уведомления могут несколько подписанных клиентов MQTT, подключенных к сети. В этом заключается большое преимущество по сравнению с традиционным способом (с использованием прикладного программного интерфейса (API) VAPIX®/ONVIF® и RTSP), при котором уведомления о событиях передаются в виде потока, поступающего в одну точку назначения.

3 Преимущества

Использование MQTT имеет ряд преимуществ. По сравнению с RTSP и другими протоколами, основанными на принципе «запрос — отклик», протокол MQTT обладает перечисленными далее преимуществами.

- **Ниже риск компрометации паролей на доступ к устройствам.** Чтобы получить данные, клиенту не нужно обращаться ни к устройству, ни к серверу. Это значит, что клиенту не нужно знать ни пароль, ни как работает API. Благодаря этому снижается риск того, что пароли на доступ к устройствам будут переданы клиентам и пользователям, а следовательно, снижается риск намеренных и случайных злоупотреблений.
- **Единая точка интеграции.** Все уполномоченные клиенты могут получать сообщения, публикуемые другими клиентами, посредством единого подключения к брокеру. В случае с RTSP клиенту необходимо подключаться к каждому клиенту, у которого необходимо получить данные. Это значит, что поток сообщений MQTT может быть организован по схеме «один — одному», «один — многим» или «многие — одному» без дополнительной нагрузки на каждый клиент.
- **Публикация и подписка через действующий межсетевой экран.** В случае с RTSP необходимо, чтобы клиенту был доступен API устройства или сервера. Если устройство находится за межсетевым экраном, а клиент является удаленным, межсетевой экран необходимо настроить на разрешение входящих запросов, лишая API защиты. Если посередине стоит брокер MQTT, то находящиеся за межсетевым экраном клиенты могут публиковать определенные данные и подписываться на них, не пробивая в экране дыру (если межсетевой экран разрешает исходящие подключения).
- **Стандарт Quality of Service.** Публикуя критически важное сообщение, публикующий может проконтролировать его получение другим клиентом, и в случае отрицательного результата совершить альтернативные действия.

- **Хранение сообщений.** Публикующие могут помечать сообщения как сохраняемые, а это значит, что брокер будет хранить копию сообщения и отправлять его вновь подключившимся клиентам, подписывающимся на соответствующую тему.
- **Доступность клиента IoT.** Существуют пакеты клиентов MQTT для всех распространенных сред разработки программного обеспечения, в том числе для Windows®, Linux®, Android™, iOS, Node.js®, PHP и Python®. К брокеру могут подключатся гораздо больше клиентов по сравнению с настройкой передачи потока данных RTSP на устройство.
- **Проще контролировать сообщения и выполнять отладку.** Имеются несколько средств MQTT, позволяющих контролировать все опубликованные сообщения, а также публиковать сообщения в целях устранения неполадок и выявления наличия и характера реакции подписчиков.
- **Более простая структура данных.** Часто планируется взаимодействие по протоколу MQTT с неизвестными клиентами — это обстоятельство обычно учитывается при формировании полезной нагрузки, чтобы упростить ее структуру для подписчика.

4 Ограничения

По сравнению с альтернативными протоколами MQTT имеет некоторые недостатки.

- **Единая точка отказа.** Если брокер окажется недоступен, перестанут работать все сообщения. Вместе с тем, в инфраструктуре можно предусмотреть резервные брокеры.
- **Кто публикует сообщение?** По замыслу разработчиков, при использовании протокола MQTT в центре внимания находится тема, а не тот, что публикует сообщения. Если в теме или в полезной нагрузке публикующий не указывает идентификатор, то чтобы узнать, кто опубликовал сообщение, необходимо обращаться к журналу брокера. Распространена практика, когда публикующие указывают тот или иной идентификатор клиента в теме или в полезной нагрузке в зависимости от сценария использования.
- **Вредоносный клиент**, подключившийся к брокеру, может публиковать данные в любых темах, к которым имеет доступ, и подписываться на такие темы. Брокер необходимо защищать (см. раздел о безопасности MQTT).
- Он **не рассчитан на непрерывную потоковую передачу ни видео-, ни аудиоматериалов.**

Как и с любым сервером, необходимо учитывать общую пропускную способность. Для очень крупных систем с большим количеством клиентов может требоваться динамическое масштабирование.

5 Инфраструктура

Довольно легко настроить локальный брокер Eclipse Mosquitto™ или задействовать Node-RED® в качестве локального брокера, такого как Aedes. Кроме того, имеется ряд поставщиков интернет-услуг и других игроков, предоставляющих управляемые брокеры MQTT, например Microsoft® Azure® IoT, HiveMQ™, CloudMQTT и IBM® Cloud®.

Если система не имеет удаленных клиентов, рекомендуется использовать локальный брокер. Локальный брокер также можно использовать как прокси-систему для общедоступного брокера или настроить на работу в качестве прокси-системы для избранных сообщений локального брокера и сообщений общедоступного брокера.

6 Безопасность

Брокеру требуется защита, соответствующая степени важности сообщений и характеру угроз, которым может подвергаться система. Протокол MQTT предлагает несколько различных схем проверки подлинности, в том числе отсутствие такой проверки, проверку по сочетанию имени пользования и пароля и проверку по сертификату клиента TLS. Пользователи могут иметь разные полномочия на публикацию сообщений на разные темы и на подписку на них. Брокер может разрешать клиентам подключаться по TCP без шифрования или по TLS с шифрованием (HTTPS и т. п.).

- **Без проверки подлинности.** Локальный брокер может выключить проверку подлинности, если сообщения не являются критически важными, а подключения к брокеру не происходят из Интернета. Такой вариант рекомендуется использовать только для тестирования, разработки в «песочнице» и демонстрации.
- **Пользователь/пароль.** Это наиболее распространенный вариант настройки. В зависимости от характера факторов риска, которым подвержена система, ее администратор может разрешить всем клиентам MQTT использовать одно сочетание имени пользователя и пароля или создать разных пользователей с ограниченным доступом к темам.
- **Сертификаты клиента TLS.** Если к брокеру происходит подключение из Интернета, а сообщения носят конфиденциальный характер, брокер необходимо настроить на допуск клиентов только по предъявлению действующего сертификата TLS. Для реализации данной схемы необходимы PKI (инфраструктура открытого ключа) и удостоверяющий центр, способный выдавать клиентские сертификаты, которым доверяет брокер. Как правило, это предлагают поставщики общедоступных интернет-услуг на базе MQTT.

В некоторых ситуациях данное решение может оказаться эффективным для сегментирования разных вариантов использования нескольких брокеров, как локальных, так и общедоступных. Выделение критических и некритических сообщений в разные сегменты является вопросом обеспечения безопасности. Применением нескольких брокеров снижается риск наличия единой точка отказа, повышается качество наблюдения и поиска и устранения неисправностей. Платой за это является развертывание и обслуживание дополнительных брокеров.

7 Клиент MQTT в устройствах Axis

В стандартной экосистеме VMS уведомления о событиях Axis, поступающие от устройств, традиционно транслируются в единую точку назначения через интерфейс API VAPIX/ONVIF с использованием протокола потоковой передачи RTSP.

Те же уведомления о событиях можно распространять при помощи протокола MQTT посредством клиента MQTT, встроенного в устройство Axis (под управлением AXIS OS 9.80 или более поздней версии). Это возможно как внутри экосистем VMS, так и за их пределами, и особенно эффективно при реализации через Интернет. Благодаря MQTT, использовать и обрабатывать уведомления о событиях, публикуемые устройством Axis, могут несколько находящихся в сети подписанных клиентов MQTT.

8 Клиенты MQTT в аналитических приложениях ACAP

Существуют приложения ACAP от Axis и сторонних поставщиков, в которых имеются собственные клиенты MQTT, разработанные для определенных систем, вариантов использования и подписчиков. Примером такого приложения является Axis Publisher — оно несет в себе дополнительные функции, структуры и поведение, необходимые некоторым системам.

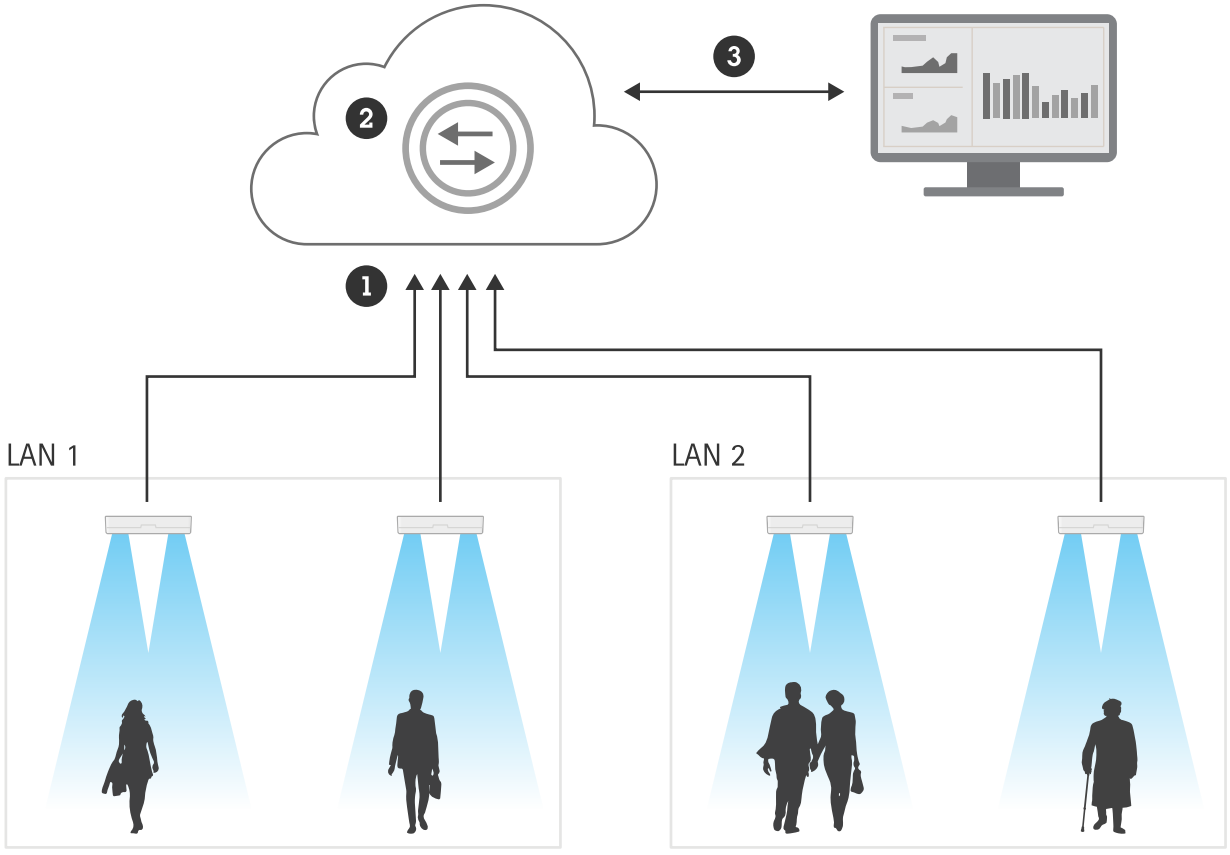
9 Другие клиенты MQTT

Имеется широкий спектр клиентов MQTT, которые устанавливаются на Linux, Windows, Android и iOS в качестве инструментов или служб для решения определенных задач. Протокол MQTT отлично подходит для реализации сценариев и применения промежуточного ПО, в том числе Node-RED/Node.js, Python и PHP. Интеграцию брокеров MQTT в свои интернет-службы прелагают Microsoft Azure IoT, AWS™, Google Cloud Platform™ и многие другие платформы. Клиент MQTT используется в датчиках, мобильных приложениях и системах (бытовой) автоматизации.

10 Практические примеры интеграции устройств с MQTT

10.1 Передача данных для анализа количества посетителей на информационную панель облачной платформы

Устройство для подсчета посетителей генерирует уведомление о событии каждый раз, когда обнаруживает, что в определенную область входит человек или выходит из нее. Уведомление передается клиенту MQTT, который в реальном времени публикует его на облачной платформе. На облачной платформе создается подключение к программному обеспечению для визуализации данных (например, к информационной панели Microsoft® Power BI®) в целях отображения оперативной статистики со счетчиков посетителей.

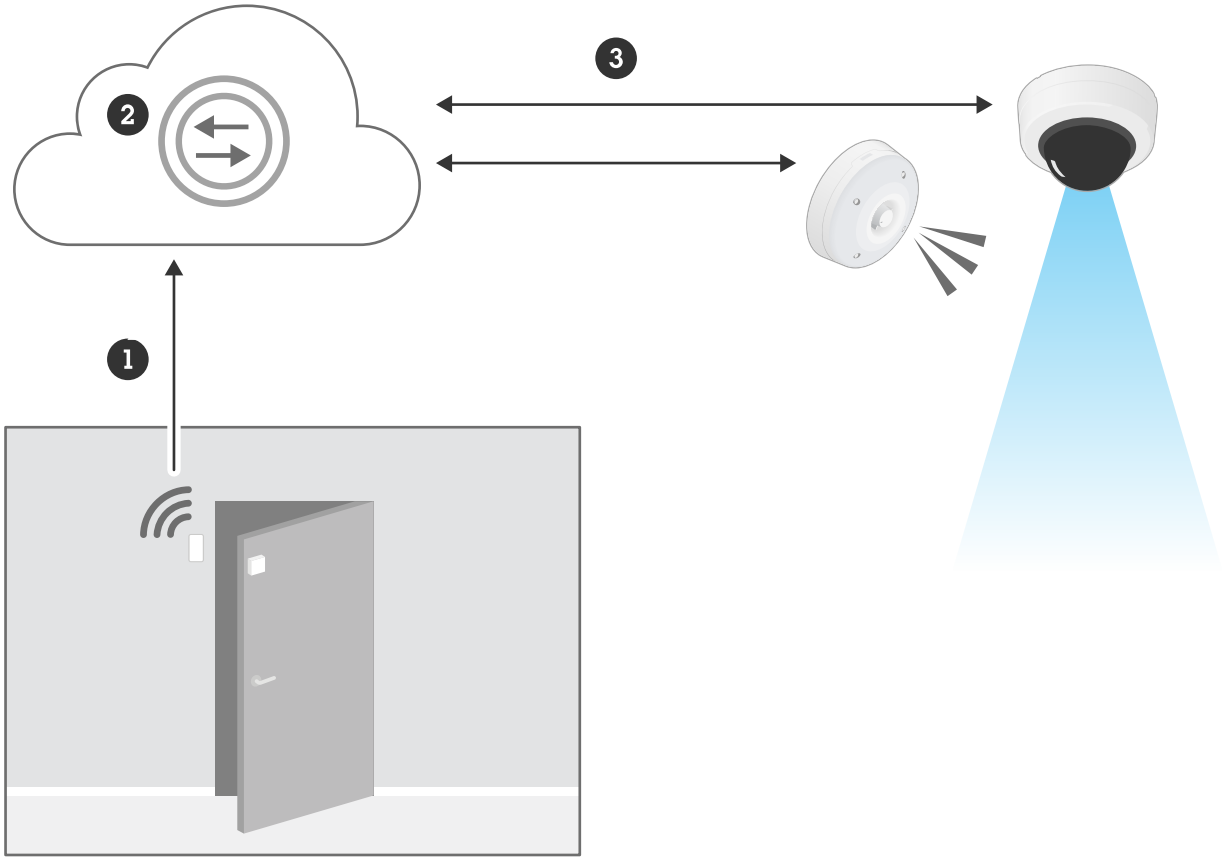


1 Публикация
2 Брокер MQTT
3 Подписка

10.2 Данные, поступающие с датчиков дверей по MQTT, инициируют срабатывание сигнализации и запуск съемки камерой

Датчик двери от стороннего производителя с поддержкой MQTT используется для инициирования уведомления о событии в случае открывания двери. Датчик двери публикует MQTT-сообщение на

брокере MQTT в облаке. Сигнальное устройство и камера подписываются на тему по датчику двери и в случае открывания двери начинают подачу сигнала тревоги и съемку соответственно.



1 *Публикация*
2 *Брокер MQTT*
3 *Подписка*

11 Словарь терминов

ACAP	*Прикладная платформа для камер AXIS*, инфраструктура для приложений, расширяющих функциональность и превносящих интеллектуальность на границе
Aedes	Брокер MQTT
API	*Прикладной программный интерфейс*, код, обеспечивающий связь между двумя программно реализованными программами
AWS™	Платформа облачных служб
AXIS OS	Операционная система для пограничных устройств от Axis
CloudMQTT	Брокер MQTT
Eclipse Mosquitto™	Брокер сообщений с открытым исходным кодом, в котором реализованы протоколы MQTT
Google Cloud Platform™	Платформа облачных служб
HiveMQ™	Брокер MQTT
HTTP;	*Протокол передачи гипертекста*, протокол передачи данных, используемый во всемирной паутине World Wide Web
IBM Cloud®	Платформа облачных служб
IoT	*Интернет вещей*, концепция интернет-подключения друг к другу вычислительных устройств, встроенных в бытовые приборы и устройства повседневного пользования
JSON	*Объектная нотация JavaScript*, компактный текстовый формат обмена данными, основанный на языке JavaScript
Microsoft® Azure® IoT	Платформа облачных служб
Microsoft® Power BI®	Интерактивная программа для визуализации данных в целях их анализа для повышения эффективности хозяйственной деятельности
MQTT	*Передача последовательности сообщений с телеметрическими данными*, протокол обмена сообщениями для Интернета вещей
Node.js®	Платформа для разработки с открытым исходным кодом, служащая для выполнения кода JavaScript на стороне сервера
Node-RED®	Средство программирования для связывания Интернета вещей
ONVIF®	Открытый отраслевой форум, занимающийся стандартизацией интерфейсов для обеспечения эффективного взаимодействия между физическими охранными устройствами на базе IP-технологий
PHP	Универсальный язык сценариев для веб-разработки
Python®	Универсальный язык программирования
RTSP	*Протокол потоковой передачи в реальном времени*, сетевой протокол для создания и контролирования медиа-сеансов между конечными точками
TCP	*Протокол управления передачей*, один из основных протоколов передачи данных в Интернете

TLS	*Безопасность на транспортном уровне*, протокол, обеспечивающий конфиденциальность и целостность данных, передаваемых по компьютерным сетям
VAPIX®	Прикладной программный интерфейс (API) для продуктов Axis
WebSocket	Коммуникационный протокол, обеспечивающий наличие каналов двусторонней связи по одному TCP-подключению
VMS	*Программное обеспечение для управления видео* или *система управления видео*

12 Принадлежность товарных знаков

Обозначения Android и Google Cloud Platform являются товарными знаками Google LLC.

Обозначение AWS является товарным знаком корпорации Amazon.com, Inc. или связанных с ней структур в США и/или других странах.

Обозначение Eclipse Mosquitto является товарным знаком Eclipse Foundation, Inc.

Обозначение HiveMQ является товарным знаком HiveMQ GmbH.

Обозначения IBM и IBM Cloud являются товарными знаками корпорации International Business Machines Corp, зарегистрированными во многих юрисдикциях по всему миру.

Обозначение IOS является товарным знаком, в том числе зарегистрированным, корпорации Cisco Systems, Inc и/или связанных с ней структур в США и ряде других стран и используется по лицензии корпорации Apple, Inc.

Обозначение JavaScript является зарегистрированным товарным знаком Oracle Corporation в США.

Обозначение Linux является зарегистрированным товарным знаком Линуса Торвальдса (Linus Torvalds) в США и других странах.

Обозначения Microsoft, Windows, Microsoft Azure IoT и Microsoft Power BI являются зарегистрированными товарными знаками Microsoft Corporation.

Обозначения Node.js и Node-RED являются зарегистрированными товарными знаками OpenJS Foundation в США и/или других странах.

Обозначение ONVIF является товарным знаком Onvif, Inc.

Обозначение Python является зарегистрированным товарным знаком Python Software Foundation.

T10162730/RU/M1.17/2203

О компании Axis Communications

Компания Axis вносит весомый вклад в формирование более разумного и безопасного мира, разрабатывая и внедряя сетевые решения, которые не только способствуют повышению безопасности, но и открывают новые пути ведения бизнеса. Занимая в отрасли ведущие позиции, компания Axis поставляет продукцию и оказывает услуги в сфере сетевого охранного видеонаблюдения и аналитики, контроля доступа, сетевых домофонов и звукового сопровождения. Свыше 3800 специалистов компании Axis трудятся более чем в 50 странах мира, вместе с нашими партнерами разрабатывая и внедряя решения стоящих перед нашими клиентами задач. Компания Axis была основана в 1984 году. Штаб-квартира компании находится в городе Лунд, Швеция.

Более подробную информацию о компании Axis можно найти на нашем веб-сайте *axis.com*.