LIVRE BLANC

Intégration des dispositifs avec le protocole MQTT

Mars 2022

AXIS COMMUNICATIONS

Avant-propos

MQTT est un protocole de messagerie normalisé qui permet l'échange efficace et fiable de données entre objets IoT et applications cloud. Il permet aux dispositifs, par l'intermédiaire de leur client MQTT, de publier des messages vers un broker (serveur) MQTT commun qui gère la communication avec les autres dispositifs. Comme le broker gère les éditeurs, la teneur de leurs messages et les demandeurs des données, il transfère uniquement les messages aux clients abonnés au sujet correspondant.

Dans un écosystème VMS type, les notifications d'événement Axis provenant des dispositifs sont automatiquement transférées vers une destination unique via l'interface API VAPIX/ONVIF par le protocole de streaming RTSP. Cependant, il est possible de distribuer ces mêmes notifications par le protocole MQTT au travers du client MQTT intégré au dispositif (pour ceux qui exécutent AXIS OS 9.80 ou une version ultérieure). Cette méthode, valable à la fois dans les écosystèmes VMS et en dehors, est particulièrement utile sur Internet. Plusieurs clients MQTT abonnés sur le réseau peuvent alors exploiter et traiter les notifications d'événement publiées par le dispositif Axis. Certaines applications d'analyse ACAP d'Axis et d'autres prestataires contiennent aussi leur propre client MQTT développé pour des systèmes, des scénarios et des abonnés particuliers.

Citons un exemple de scénario lié aux produits Axis : les dispositifs de comptage de personnes peuvent envoyer des statistiques par MQTT à des logiciels de visualisation de données dans le cloud. Autre exemple : un capteur de porte d'un autre fournisseur communique par MQTT avec un dispositif de signalisation et une caméra, qui émet une alarme et lance un enregistrement à chaque ouverture de la porte.

Table des matières

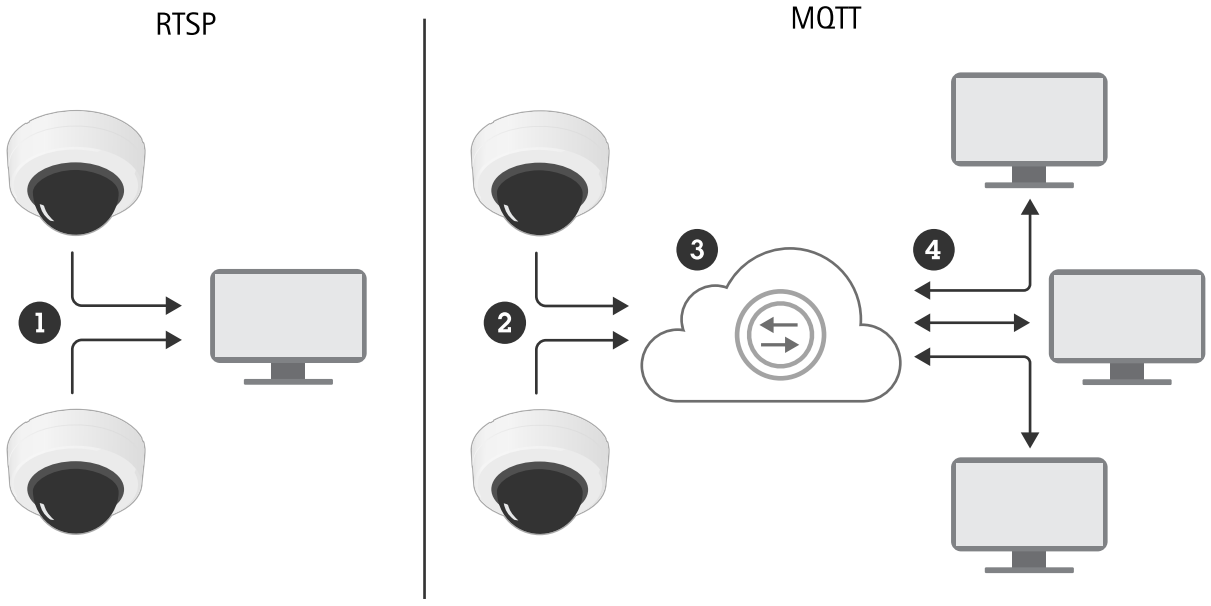
1 Introduction

MQTT (Message Queuing Telemetry Transport) est un protocole de messagerie normalisé pour l'Internet des objets (IoT). Conçu pour simplifier l'intégration IoT, il est utilisé dans de nombreux secteurs pour connecter des dispositifs distants avec une empreinte de code réduite et une bande passante réseau minimale. Le client MQTT d'AXIS OS peut simplifier l'intégration des données et des événements produits sur le dispositif aux systèmes autres que les systèmes de gestion vidéo (VMS).

Ce livre blanc présente un tour d'horizon technique du protocole MQTT, accompagné de scénarios d'utilisation typiques et d'un panorama des avantages et des limitations. Il donne également des détails sur les clients MQTT des dispositifs Axis et les applications d'analyse ACAP.

2 Protocole MQTT

MQTT est un protocole de publication/abonnement. De ce fait, il possède un modèle de message différent de celui des protocoles RTSP ou HTTP, qui sont des protocoles de demande/réponse. Avec RTSP, une extrémité émet une demande et l'autre répond. De nombreuses applis de messagerie mobile sont au contraire basées sur MQTT ou des concepts similaires de publication/abonnement. Il existe également des protocoles de publication/abonnement optimisés pour des systèmes fermés particuliers.



Alors que le protocole RTSP se limite aux communications un-à-un, MQTT permet également les communications un-à-plusieurs et plusieurs-à-plusieurs par l'intermédiaire du broker.

1 *Flux d'événements*
2 *Publication*
3 *Broker MQTT*
4 *Abonnement*

Le concept MQTT consiste à connecter tous les clients à un broker (serveur) MQTT, qui gère les éditeurs et les abonnés qui veulent recevoir les données. La connexion est généralement une session TCP sur le port 1883. Un client peut également se connecter par TLS (port standard 8883) ou avec WebSocket (port standard 1884/8884).

Les clients publient des messages avec un sujet. Un autre client peut s'abonner à ce sujet ou utiliser des caractères génériques pour récupérer tous les sous-sujets. Un message contient également une charge utile, généralement une structure de données JSON, une chaîne de caractères ou même quelques données binaires. L'éditeur ne sait pas si d'autres clients s'abonnent. Le broker transmet les messages uniquement aux clients qui possèdent un abonnement au sujet.

MQTT revient en quelque sorte à envoyer un article à un magazine. Les personnes qui s'abonnent au magazine peuvent lire l'article, et la communication peut être soit de type un-à-un, soit un-à-plusieurs (MQTT permet même les communications plusieurs-à-plusieurs). L'article reste également lisible longtemps après sa première publication.

Pour poursuivre la comparaison, RTSP est plutôt analogue à un appel téléphonique. Il existe une seule source et une seule cible pour vos commandes, et la communication est toujours un-à-un. Si votre cible ne répond pas au téléphone, elle ne reçoit pas le message.

Lorsque MQTT sert à diffuser des notifications d'événement Axis à partir des dispositifs, plusieurs clients MQTT abonnés du réseau peuvent utiliser et traiter ces notifications. Cet avantage est considérable par rapport à la méthode traditionnelle (utilisant l'API VAPIX®/ONVIF® et RTSP), où les notifications d'événement sont au contraire transmises à une seule destination.

3 Avantages

Par rapport à un protocole de demande/réponse comme RTSP, MQTT présente plusieurs avantages :

- **Moindre risque d'exposition des mots de passe des dispositifs.** Les clients n'ont pas besoin d'accéder au dispositif ou au serveur pour récupérer les données. Concrètement, le client n'a pas besoin de connaître le mot de passe ou le fonctionnement de l'API. De ce fait, les mots de passe des dispositifs ont moins de risque de fuiter vers les clients et les utilisateurs, limitant du même coup le risque de détournement accidentel ou délibéré.
- **Point unique d'intégration.** S'ils sont autorisés, tous les clients peuvent obtenir les messages publiés de tous les autres clients avec une seule connexion à un broker. Avec RTSP, un client doit se connecter à chaque client dont il souhaite récupérer les données. Par conséquent, le flux des messages MQTT peut être de type un-à-un, un-à-plusieurs ou plusieurs-à-un, sans charge supplémentaire pour chaque client.
- **Publication et abonnement sans ouverture du pare-feu** Avec RTSP, l'API du dispositif/serveur doit être accessible au client. Si le dispositif est derrière un pare-feu et le client est distant, le pare-feu doit être configuré pour autoriser les demandes entrantes, ce qui expose l'API du dispositif. Avec un broker MQTT public entre les deux, les clients derrière un pare-feu peuvent publier/s'abonner à des données spécifiques sans avoir à ouvrir de brèche dans le pare-feu (si le pare-feu autorise les connexions sortantes).
- **Qualité de service.** Lorsqu'un éditeur publie un message critique, il peut vérifier s'il est reçu par un autre client et, dans la négative, exécuter une autre action.
- **Messages retenus.** Les éditeurs peuvent marquer un message comme retenu, c'est-à-dire que le broker conserve une copie du message et l'envoie aux clients nouvellement connectés qui s'abonnent à ce sujet.
- **Disponibilité des clients IoT.** Les logiciels clients MQTT sont disponibles pour la plupart des environnements courants de développement logiciel, notamment Windows®, Linux®, Android™, iOS, Node.js®, PHP et Python®. Par rapport à la configuration d'un flux de données RTSP vers un dispositif, beaucoup d'autres clients peuvent se connecter à un broker.

- **Contrôle et débogage simplifiés des messages.** Il existe plusieurs outils MQTT permettant de surveiller tous les messages publiés et de publier des messages de dépannage pour déterminer si les abonnés réagissent et comment.
- **Structure simplifiée des données.** Comme MQTT cible souvent des clients inconnus, la charge utile du message en tient compte pour simplifier les données destinées à l'abonné.

4 Limites

MQTT présente certains inconvénients par rapport à d'autres protocoles :

- **Point unique de défaillance.** Si le broker n'est pas disponible, plus aucun message n'est transmis. Cependant, il est possible de concevoir l'infrastructure avec des brokers redondants.
- **Éditeur des messages.** Par principe, MQTT est centré sur le sujet et non sur l'éditeur du message. Si l'éditeur n'inclut pas une forme d'identifiant dans le sujet ou la charge utile, vous devez consulter le journal du broker pour le déterminer. Néanmoins, les éditeurs incluent souvent une forme d'ID client dans le sujet ou la charge utile d'après le scénario d'utilisation.
- **Un client malveillant** connecté au broker peut publier ou s'abonner à tous les sujets pour lesquels il est autorisé. Vous devez donc protéger le broker (voir le chapitre sur la sécurité MQTT).
- Il n'est **pas conçu pour le streaming vidéo/audio en continu.**

Comme pour tous les serveurs, il faut prendre en compte les besoins totaux en bande passante. Pour les très grands systèmes comptant de nombreux clients, une adaptation de charge dynamique peut s'avérer nécessaire.

5 Infrastructure

Il est facile de configurer un broker local Eclipse Mosquitto™ ou d'activer Node-RED® pour qu'il agisse comme broker local, par exemple Aedes. Il existe également plusieurs fournisseurs de services Internet et d'autres qui proposent des brokers MQTT gérés, comme Microsoft® Azure® IoT, HiveMQ™, CloudMQTT et IBM® Cloud®.

Si le système ne comporte pas de clients distants, il est conseillé d'utiliser un broker local. Un broker local peut également faire office de proxy vers un broker public ou être configuré comme proxy pour les messages sélectionnés du broker local et les messages du broker public.

6 Sécurité

Le broker a besoin d'un niveau de protection en accord avec le degré de sensibilité des messages et le type de menace que peut craindre un système particulier. MQTT propose plusieurs méthodes d'authentification : sans authentification, utilisateur/mot de passe et authentification par certificat client TLS. Un utilisateur peut disposer d'autorisations différentes selon le sujet qu'il publie ou auquel il s'abonne. Le broker peut autoriser les clients à se connecter par TCP non crypté ou TLS crypté (par ex. HTTPS).

- **Pas d'authentification.** Un broker local peut désactiver l'authentification si les messages ne sont pas critiques et si le broker n'est pas exposé pour les clients Internet. Il est recommandé d'employer cette méthode uniquement pour les tests, le développement en bac à sable et les démonstrations.

- **Utilisateur/mot de passe.** Cette configuration est la plus courante. En fonction des risques auxquels est exposé le système, l'administrateur système peut autoriser tous les clients MQTT à utiliser les mêmes identifiants utilisateur/mot de passe ou créer des utilisateurs autorisés à accéder à des sujets restreints.
- **Certificats clients TLS.** Pour les brokers exposés à Internet dont les messages sont classés sensibles, il convient de les configurer pour autoriser seulement les clients disposant d'un certificat TLS valide. Cette méthode nécessite une infrastructure à clé publique (PKI) et une autorité de certification capable de délivrer des certificats clients approuvés par le broker. En général, les prestataires Internet de services MQTT publics la proposent.

Dans certains cas, il peut être judicieux de segmenter les différents scénarios d'utilisation sur plusieurs brokers locaux et/ou publics. La segmentation entre messages critiques et non critiques constitue un contrôle de sécurité. Le choix de plusieurs brokers réduit également le risque de point unique de défaillance tout en renforçant le contrôle et la résolution des problèmes. Le coût supplémentaire porte sur le déploiement et la maintenance de brokers additionnels.

7 Client MQTT dans les dispositifs Axis

Dans un écosystème VMS standard, les notifications d'événement Axis provenant des dispositifs sont automatiquement transférées vers une destination unique via l'interface API VAPIX/ONVIF par le protocole de streaming RTSP.

Il est possible de distribuer ces mêmes notifications d'événement avec le protocole MQTT par l'intermédiaire du client MQTT intégré à un dispositif Axis (qui exécute la version AXIS OS 9.80 ou ultérieure). Cette méthode, valable à la fois dans les écosystèmes VMS et en dehors, est particulièrement utile sur Internet. Avec MQTT, plusieurs clients MQTT abonnés sur le réseau peuvent utiliser et traiter les notifications d'événement publiées par le dispositif Axis.

8 Clients MQTT dans les applications d'analyse ACAP

Certaines applications ACAP d'Axis et d'autres prestataires contiennent leurs propres clients MQTT développés pour des systèmes, des scénarios et des abonnés particuliers. Par exemple, Axis Publisher ajoute des fonctionnalités, des structures et des comportements nécessaires à certains systèmes.

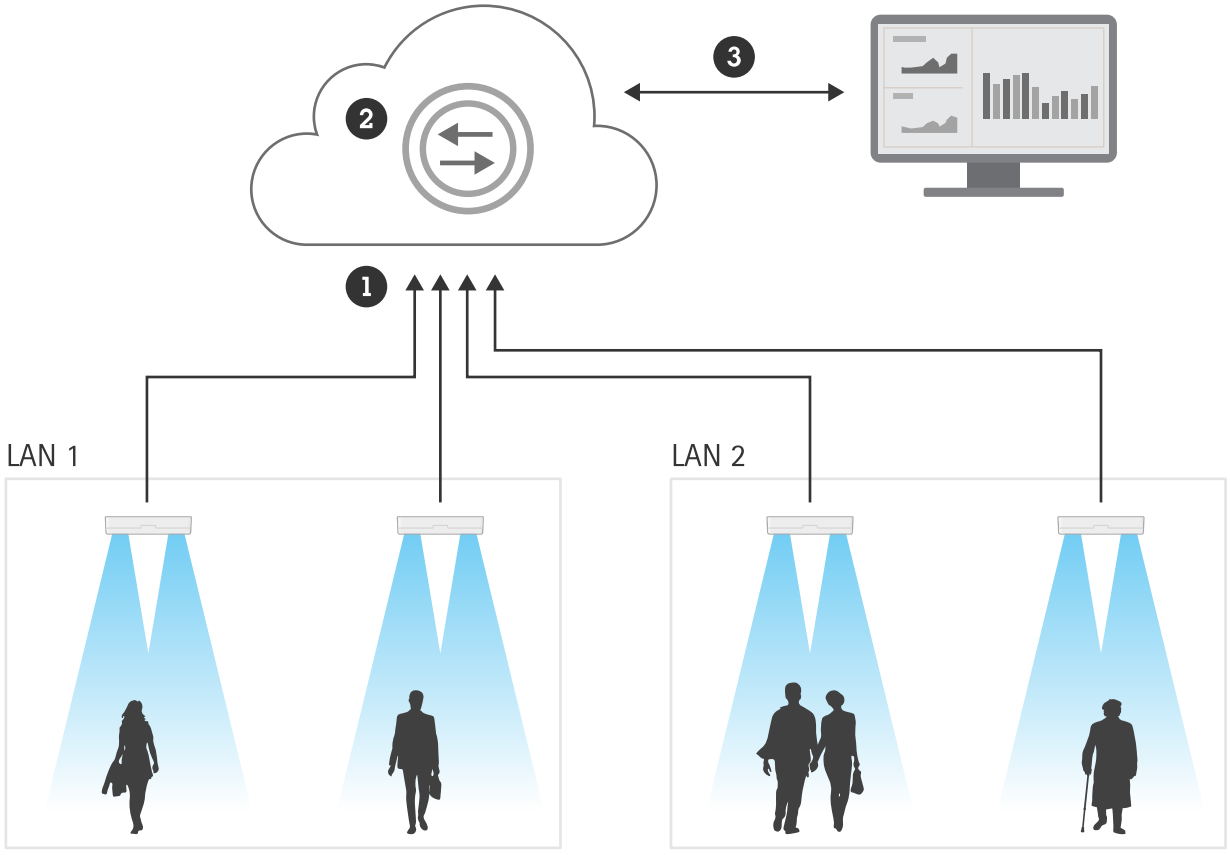
9 Autres clients MQTT

Une grande diversité de clients MQTT peuvent être installés sur Linux, Windows, Android et iOS. Ils se présentent sous forme d'outils ou de services pour des scénarios particuliers. MQTT convient parfaitement pour les outils de développement et de script comme Node-RED/Node.js, Python et PHP. La plupart des plateformes de services Internet, dont Microsoft Azure IoT, AWS™ ou Google Cloud Platform™, proposent des brokers MQTT à intégrer aux services exécutés sur la plateforme. Des capteurs, applications mobiles et systèmes domotiques possèdent un client MQTT.

10 Exemples de scénarios d'intégration de dispositifs avec MQTT

10.1 Données d'analyse du comptage de personnes vers le tableau de bord d'une plateforme cloud

Un dispositif doté de fonctions de comptage de personnes génère une notification d'événement à chaque franchissement « entrant » ou « sortant » détecté dans une zone définie. La notification est transmise au client MQTT qui la publie en temps réel sur la plateforme cloud. Sur la plateforme cloud, une connexion est établie avec un logiciel de visualisation des données (par exemple un tableau de bord Microsoft® Power BI®) pour afficher les statistiques en temps réel des compteurs de personnes.

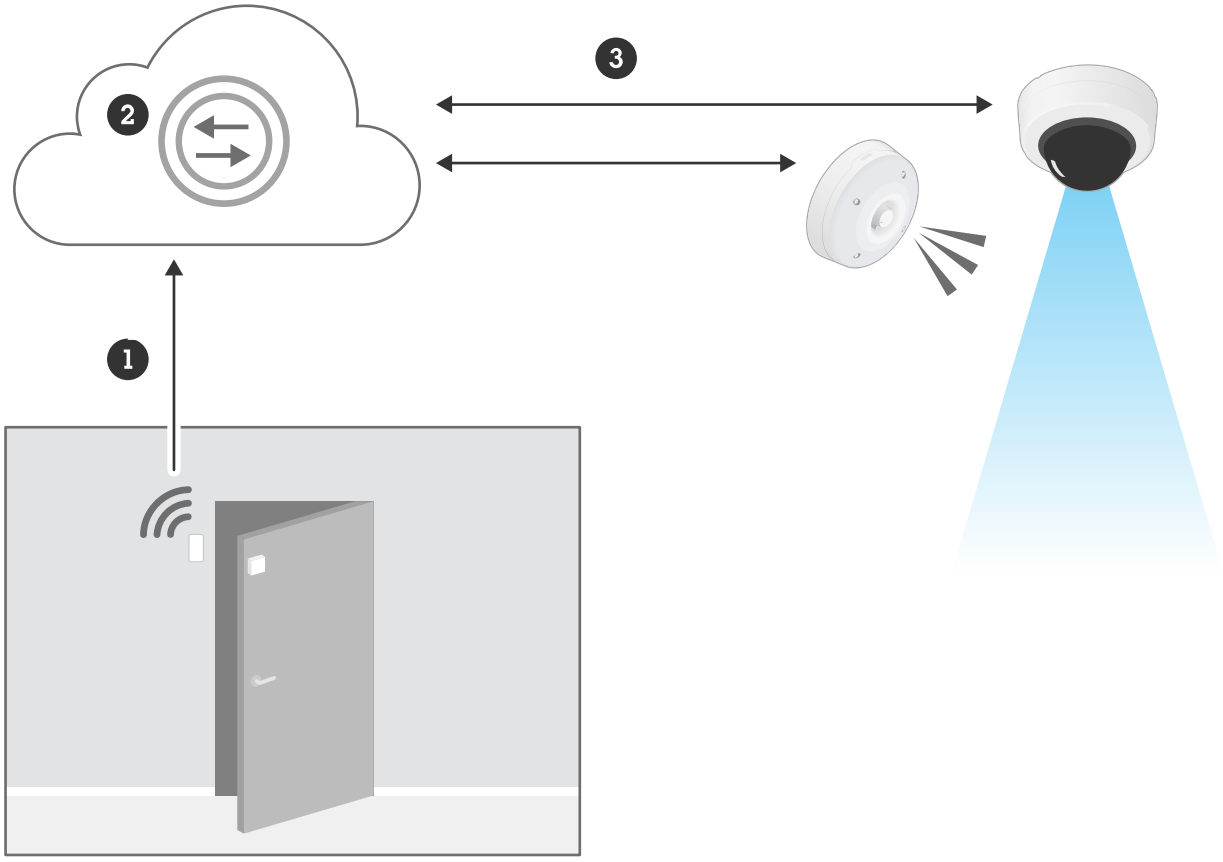


1 *Publication*
2 *Broker MQTT*
3 *Abonnement*

10.2 Données d'un capteur de porte par MQTT qui déclenchent une alarme du dispositif de signalisation et un enregistrement par la caméra

Un capteur de porte MQTT d'un autre fournisseur sert à déclencher une notification d'événement en cas d'ouverture de la porte. Le capteur de porte publie un message MQTT au broker MQTT dans le cloud. Le

dispositif de signalisation et la caméra sont abonnés au sujet du capteur de porte ; si la porte s'ouvre, l'un émet une alarme tandis que l'autre démarre un enregistrement.



1 *Publication*
2 *Broker MQTT*
3 *Abonnement*

11 Glossaire

ACAP	*AXIS Camera Application Platform*, cadre de développement d'applications qui enrichissent les fonctionnalités et l'analyse en périphérie de réseau
Aedes	Broker MQTT
API	*Application Programming Interface*, code permettant à deux logiciels de communiquer entre eux
AWS™	Nom d'une plateforme de services cloud
AXIS OS	Système d'exploitation des dispositifs Axis en périphérie de réseau
CloudMQTT	Broker MQTT
Eclipse Mosquitto™	Broker de messages en source ouverte qui applique les protocoles MQTT
Google Cloud Platform™	Nom d'une plateforme de services cloud
HiveMQ™	Broker MQTT
HTTP	*HyperText Transfer Protocol*, protocole de transfert de données utilisé sur le web
IBM Cloud®	Nom d'une plateforme de services cloud
IoT	*Internet of Things* ou Internet des objets, interconnexion par Internet des dispositifs informatiques intégrés aux appareils et aux capteurs du quotidien
JSON	*JavaScript Object Notation*, format de fichier compact et format d'échange de données
Microsoft® Azure® IoT	Nom d'une plateforme de services cloud
Microsoft® Power BI®	Logiciel interactif de visualisation des données axé sur la Business Intelligence
MQTT	*Message Queuing Telemetry Transport*, protocole de messagerie pour l'Internet des objets (IoT).
Node.js®	Plateforme de développement en source ouverte pour l'exécution de code JavaScript côté serveur
Node-RED®	Outil de programmation pour mailler l'Internet des objets
ONVIF®	Forum open source assurant la fourniture et la promotion d'interfaces normalisées pour une interopérabilité efficace des produits de sécurité physique sur IP
PHP	Langage de script général axé sur le développement web
Python®	Langage de programmation général
RTSP	*Real-Time Streaming Protocol*, protocole réseau permettant d'établir et de gérer les sessions multimédia entre terminaux
TCP	*Transmission Control Protocol*, un des principaux protocoles de transmission de données par Internet
Spécification de haut niveau	*Transport Layer Security*, protocole assurant la confidentialité et l'intégrité des communications sur les réseaux informatiques
VAPIX®	Interface de programmation d'applications (API) ouverte pour les produits Axis

WebSocket	Protocole de communication assurant des canaux de communication bidirectionnelle sur une seule connexion TCP
VMS	*Video Management Software*, logiciel de gestion vidéo

12 Attribution des marques commerciales

Android et Google Cloud Platform sont des marques commerciales de Google LLC.

AWS est une marque commerciale d'Amazon.com, Inc. ou de ses filiales aux États-Unis et/ou dans d'autres pays.

Eclipse Mosquitto est une marque commerciale d'Eclipse Foundation, Inc.

HiveMQ est une marque commerciale de HiveMQ GmbH.

IBM et IBM Cloud sont des marques commerciales d'International Business Machines Corp, enregistrées dans de nombreuses juridictions dans le monde.

IOS est une marque commerciale ou marque déposée de Cisco Systems, Inc et/ou de ses filiales aux États-Unis et dans certains autres pays, qui est exploitée sous licence par Apple, Inc.

JavaScript est une marque déposée d'Oracle Corporation aux États-Unis.

Linux est une marque déposée par Linus Torvalds aux États-Unis et dans d'autres pays.

Microsoft, Windows, Microsoft Azure IoT et Microsoft Power BI sont des marques déposées de Microsoft Corporation.

Node.js et Node-RED sont des marques déposées de l'OpenJS Foundation aux États-Unis et/ou dans d'autres pays.

ONVIF est une marque commerciale d'Onvif, Inc.

Python est une marque déposée de la Python Software Foundation.

T10162730/FR/M1.17/2203

À propos d'Axis Communications

En concevant des solutions qui améliorent la sécurité et les performances de l'entreprise, Axis crée monde plus clairvoyant et plus sûr. En tant qu'entreprise de technologie de réseau et leader de l'industrie, Axis propose des solutions de vidéosurveillance, de contrôle d'accès, d'interphonie et de systèmes audio. Les performances de ces solutions sont améliorées grâce à des applications d'analyse intelligentes et une formation de haute qualité.

Axis emploie près de 4 000 personnes dans plus de 50 pays et collabore avec des partenaires technologiques et d'intégration de systèmes dans le monde entier pour fournir des solutions clients adaptées. Axis a été fondée en 1984 et le siège social se trouve à Lund, en Suède