WHITE PAPER

Integrazione dei dispositivi con MQTT

Marzo 2022

AXIS COMMUNICATIONS

Sommario

MQTT è un protocollo di messaggistica standard che agevola uno scambio di dati efficiente e affidabile tra dispositivi IoT e applicazioni cloud. Consente ai dispositivi (attraverso i loro client MQTT) di pubblicare messaggi su un broker MQTT comune (server) che media la comunicazione con altri dispositivi. Il broker monitora chi sta pubblicando un messaggio (e i relativi contenuti) e chi desidera vedere i dati, inoltrando i messaggi solo ai client che si iscrivono all'argomento giusto.

In un ecosistema VMS tipico, le notifiche di eventi Axis dai dispositivi vengono in genere trasmesse a un'unica destinazione tramite l'interfaccia API VAPIX/ONVIF utilizzando il protocollo di streaming RTSP. Tuttavia, le stesse notifiche possono essere distribuite con il protocollo MQTT tramite il client MQTT integrato nel dispositivo (solo per i dispositivi che eseguono AXIS OS 9.80 o una versione successiva). Questo è possibile sia all'interno degli ecosistemi VMS che al di fuori ed è particolarmente utile su Internet. Più client MQTT sottoscritti nella rete possono quindi utilizzare ed elaborare le notifiche di eventi pubblicate dal dispositivo Axis. Sono anche disponibili app Axis e applicazioni analitiche ACAP di terze parti che dispongono di client MQTT propri, progettati per sistemi, applicazioni e subscriber specifici.

Come esempio di applicazione relativo ai prodotti Axis, i dispositivi di conteggio persone possono inviare statistiche tramite MQTT al software di visualizzazione dei dati nel cloud. Altro esempio: un sensore per porta di terze parti comunica tramite MQTT con un dispositivo di segnalazione e una telecamera, che emettono un allarme e avviano una registrazione ad ogni apertura della porta.

Sommario

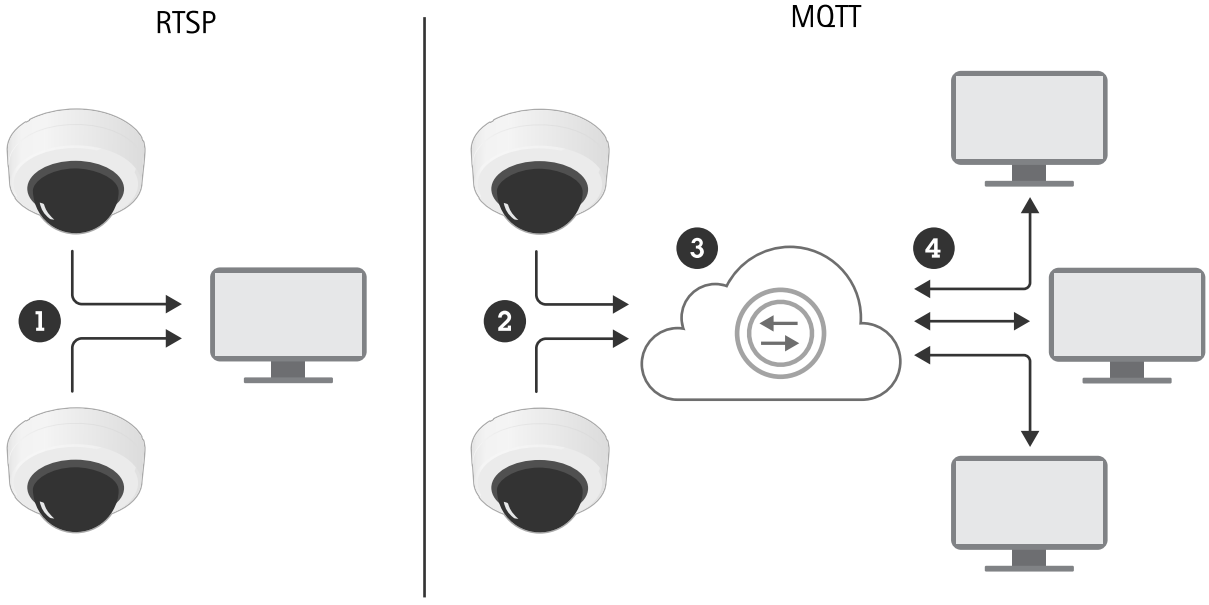
1 Introduzione

MQTT (Message Queuing Telemetry Transport) è un protocollo di messaggistica standard per l'Internet of Things (IoT). È stato progettato per un'integrazione IoT semplificata ed è utilizzato in numerosi settori per connettere dispositivi remoti con un'impronta di codice ridotta e una larghezza di banda minima in rete. Il client MQTT in AXIS OS può semplificare l'integrazione di dati ed eventi prodotti dal dispositivo con sistemi diversi dai quelli di gestione video (VMS).

Questo documento tecnico illustra le nozioni di base su MQTT, come le applicazioni possibili, i vantaggi e le limitazioni. Inoltre, fornisce dettagli sui client MQTT dei dispositivi Axis e delle app analitiche ACAP.

2 Il protocollo MQTT

MQTT è un protocollo di tipo publish/subscribe (pubblicazione/sottoscrizione). Questo significa che ha un modello di messaggistica diverso da RTSP e HTTP, che sono protocolli request/response (richiesta/risposta). Con RTSP, una parte invia una richiesta e l'altra risponde. Molte app di messaggistica per dispositivi mobili si basano invece su MQTT o metodi simili di tipo publish/subscribe. Esistono anche protocolli publish/subscribe ottimizzati per sistemi chiusi o specifici.



Mentre RTSP consente solo la comunicazione uno a uno, MQTT consente anche la comunicazione uno a molti o molti a molti tramite il broker.

1 *Flusso degli eventi*
2 *Publish*
3 *Broker MQTT*
4 *Subscribe*

Il concetto di MQTT è che tutti i client si connettono a un broker MQTT comune (server), che monitora chi sta pubblicando un messaggio (e i relativi contenuti) e chi desidera vedere i dati. Normalmente, la connessione avviene tramite una sessione TCP sulla porta 1883. Un client può anche connettersi tramite TLS (in genere sulla porta 8883) o WebSocket (in genere sulla porta 1884/8884).

I client pubblicano messaggi con un argomento. Un altro client può iscriversi a quell'argomento specifico o utilizzare caratteri jolly per ottenere tutti i sottoargomenti. Un messaggio include anche un payload, che è in genere una struttura dati JSON, una stringa o anche una breve sequenza di dati binari. Il publisher non sa se altri client siano iscritti. Il broker inoltra i messaggi solo ai client che sottoscrivono gli argomenti.

MQTT è un po' come inviare un articolo a una rivista. Gli abbonati alla rivista possono leggere l'articolo e la comunicazione può essere uno a uno o uno a molti (MQTT consente anche la comunicazione molti a molti). L'articolo può anche essere letto molto tempo dopo la pubblicazione.

RTSP è invece più simile a una telefonata. I comandi hanno un'origine e una destinazione e la comunicazione è sempre uno a uno. Se il destinatario non risponde al telefono, il messaggio è perso.

Utilizzando MQTT per distribuire le notifiche di eventi Axis dai dispositivi, più client MQTT sottoscritti nella rete possono utilizzare ed elaborare le notifiche. Si tratta di un grande vantaggio rispetto al metodo tradizionale (tramite API VAPIX®/ONVIF® e RTSP), che trasmette le notifiche di eventi a un'unica destinazione.

3 Vantaggi

Rispetto a un protocollo di richiesta/risposta come RTSP, l'uso di MQTT offre diversi vantaggi. Tra i principali:

- **Minore rischio di esposizione delle password dei dispositivi.** Non è necessario che un client acceda a un dispositivo o a un server per ottenere i dati. Questo significa che al client non serve conoscere la password né il funzionamento dell'API. In questo modo, si rischia meno che le password dei dispositivi vengano esposte a client e utenti, riducendo il rischio di un uso improprio intenzionale o accidentale.
- **Singolo punto di integrazione.** Se autorizzati, tutti i client possono ottenere i messaggi pubblicati da tutti gli altri client con un'unica connessione al broker. Con RTSP, un client deve connettersi a ogni client da cui desidera i dati. Questo significa che il flusso di messaggi MQTT può essere uno a uno, uno a molti o molti a uno senza carichi aggiuntivi per ogni client.
- **Pubblicazione/sottoscrizione con firewall intatto** In RTSP, l'API del dispositivo/server deve essere accessibile al client. Se il dispositivo è protetto da un firewall e il client è remoto, il firewall deve essere configurato per consentire le richieste in entrata, esponendo l'API del dispositivo. Con un broker MQTT pubblico in posizione intermedia, i client protetti da firewall possono pubblicare/sottoscrivere dati specifici senza aprire un varco nel firewall (se il firewall consente le connessioni in uscita).
- **Qualità del servizio.** Quando pubblica un messaggio critico, un publisher può monitorare l'avvenuta ricezione del messaggio da un altro client e, in caso contrario, intraprendere un'azione alternativa.
- **Messaggi conservati.** I publisher possono contrassegnare un messaggio come conservato; ovvero, il broker conserva una copia del messaggio e lo invia ai nuovi client connessi che si iscrivono a tale argomento.
- **Disponibilità client IoT.** I pacchetti client MQTT sono disponibili per tutti gli ambienti di sviluppo software più comuni, come Windows®, Linux®, Android™, iOS, Node.js®, PHP e Python®. Esistono molti altri client che possono connettersi a un broker rispetto alla configurazione di un flusso di dati RTSP su un dispositivo.
- **Monitoraggio e debug semplificati dei messaggi.** È possibile utilizzare diversi strumenti MQTT per monitorare tutti i messaggi pubblicati e per pubblicare messaggi risolutivi in base all'eventuale reazione (e alla modalità di reazione) dei subscriber.
- **Struttura dei dati semplificata.** Poiché MQTT si rivolge spesso a client sconosciuti, in genere il payload del messaggio ne tiene conto per semplificare le operazioni al subscriber.

4 Limitazioni

MQTT ha alcuni svantaggi rispetto ai protocolli alternativi:

- **Singolo punto di errore.** Se il broker non è disponibile, tutte le attività di messaggistica smettono di funzionare. Tuttavia, l'infrastruttura può essere progettata con broker ridondanti.
- **Chi ha pubblicato un messaggio?** Per caratteristiche intrinseche di progettazione, MQTT pone l'attenzione sull'argomento e non su chi ha pubblicato il messaggio. A meno che il publisher non includa un ID nell'argomento o nel payload, è necessario accedere al registro del broker per sapere chi ha pubblicato il messaggio. Per prassi comune, i publisher includono alcuni ID client nell'argomento o nel payload in base al caso d'uso.
- **Un client dannoso** connesso al broker può pubblicare/iscriversi a qualsiasi argomento per il quale è autorizzato. È necessario proteggere il broker (vedere la sezione sulla sicurezza di MQTT).
- **Non è progettato per lo streaming video/audio continuo.**

Come con qualsiasi server, è necessario considerare la larghezza di banda totale utilizzata. Per sistemi molto grandi con molti client, può essere necessario il ridimensionamento dinamico.

5 Infrastruttura

È piuttosto semplice configurare un broker Eclipse Mosquitto™ locale o abilitare Node-RED® affinché funzioni come un broker locale, ad esempio Aedes. Esistono anche numerosi provider di servizi Internet e altro che forniscono broker MQTT gestiti, come Microsoft® Azure® IoT, HiveMQ™, CloudMQTT e IBM® Cloud®.

Se un sistema non ha client remoti, si consiglia di utilizzare un broker locale. Un broker locale può anche fungere da proxy per un broker pubblico, oppure essere configurato per agire da proxy per alcuni messaggi del broker locale e per i messaggi del broker pubblico.

6 Sicurezza

Il broker necessita di una protezione adeguata in base alla criticità dei messaggi e alle minacce che un sistema specifico potrebbe subire. MQTT offre diversi metodi di autenticazione, come nessuna autenticazione, utente/password e autenticazione del certificato client TLS. Utenti diversi possono avere autorizzazioni diverse in base all'argomento da pubblicare o al quale iscriversi. Il broker può consentire ai client di connettersi tramite TCP non crittografato o TLS crittografato (come HTTPS).

- **Nessuna autenticazione.** Un broker locale può disabilitare l'autenticazione se i messaggi non sono critici e il broker non è esposto ai client Internet. Si consiglia di utilizzarlo solo per test, sviluppo di sandbox e dimostrazioni.
- **Utente/password.** Si tratta della configurazione più comune. A seconda dei rischi del sistema, l'amministratore può consentire a tutti i client MQTT di condividere lo stesso utente/password o creare utenti con accesso limitato a seconda degli argomenti.
- **Certificati client TLS.** Per i broker esposti a Internet in cui i messaggi sono classificati come sensibili, il broker deve essere configurato per ammettere solo client che dispongano di un certificato TLS valido. Questo metodo richiede una PKI (infrastruttura a chiave pubblica) e un'autorità di certificazione in grado di emettere certificati client attendibili per il broker. In genere, i provider di servizi Internet MQTT pubblici offrono questa possibilità.

In alcune situazioni, può essere opportuno segmentare applicazioni diverse con più broker, che possono essere locali e/o pubblici. La segmentazione dei messaggi critici e non critici è un controllo di sicurezza. La presenza di più broker riduce anche il rischio di singolo punto di errore e migliora il monitoraggio e la risoluzione dei problemi. Il prezzo da pagare è l'implementazione aggiuntiva e la manutenzione di broker supplementari.

7 Client MQTT nei dispositivi Axis

In un ecosistema VMS standard, le notifiche di eventi Axis dai dispositivi vengono in genere trasmesse a un'unica destinazione tramite l'interfaccia API VAPIX/ONVIF utilizzando il protocollo di streaming RTSP.

Le stesse notifiche di eventi possono essere distribuite utilizzando il protocollo MQTT tramite il client MQTT integrato di un dispositivo Axis (che esegue AXIS OS 9.80 o una versione successiva). Questo vale sia all'interno degli ecosistemi VMS che al di fuori ed è particolarmente utile su Internet. Con MQTT, più client MQTT sottoscritti nella rete possono utilizzare ed elaborare le notifiche di eventi pubblicate dal dispositivo Axis.

8 Client MQTT nelle applicazioni analitiche ACAP

Sono disponibili app Axis e ACAP di terze parti che dispongono di client MQTT propri, progettati per sistemi, applicazioni e subscriber specifici. Axis Publisher è un esempio che aggiunge funzionalità, strutture e comportamenti aggiuntivi necessari per alcuni sistemi.

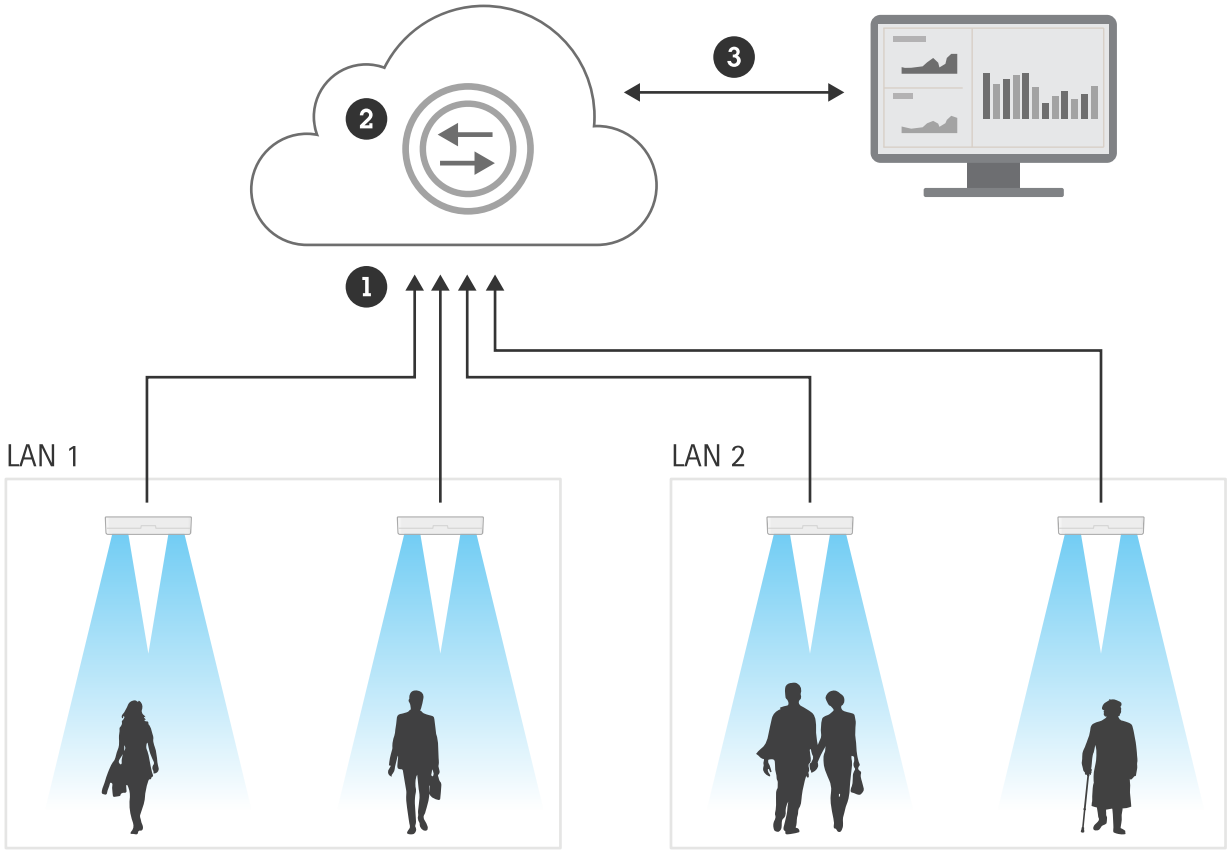
9 Altri client MQTT

Esiste un'ampia gamma di client MQTT che possono essere installati su Linux, Windows, Android e iOS, progettati come strumenti o servizi per applicazioni specifiche. MQTT è molto adatto a script e middleware come Node-RED/Node.js, Python e PHP. La maggior parte delle piattaforme di servizi Internet come Microsoft Azure IoT, AWS™ e Google Cloud Platform™ offre broker MQTT da integrare nei servizi in esecuzione sulla piattaforma. Esistono sensori, app per dispositivi mobili e sistemi di automazione (domotica) dotati di un client MQTT.

10 Esempi di applicazioni che integrano dispositivi con MQTT

10.1 Dati analitici di conteggio persone sul dashboard della piattaforma cloud

Un dispositivo con analitica di conteggio persone genera una notifica di evento ogni volta che rileva una persona che entra o esce da un'area prestabilita. La notifica viene trasmessa al client MQTT, che la pubblica in tempo reale sulla piattaforma cloud. Sulla piattaforma cloud viene creata una connessione al software di visualizzazione dati (ad esempio un dashboard Microsoft® Power BI®) per visualizzare le statistiche in tempo reale dei contapersone.

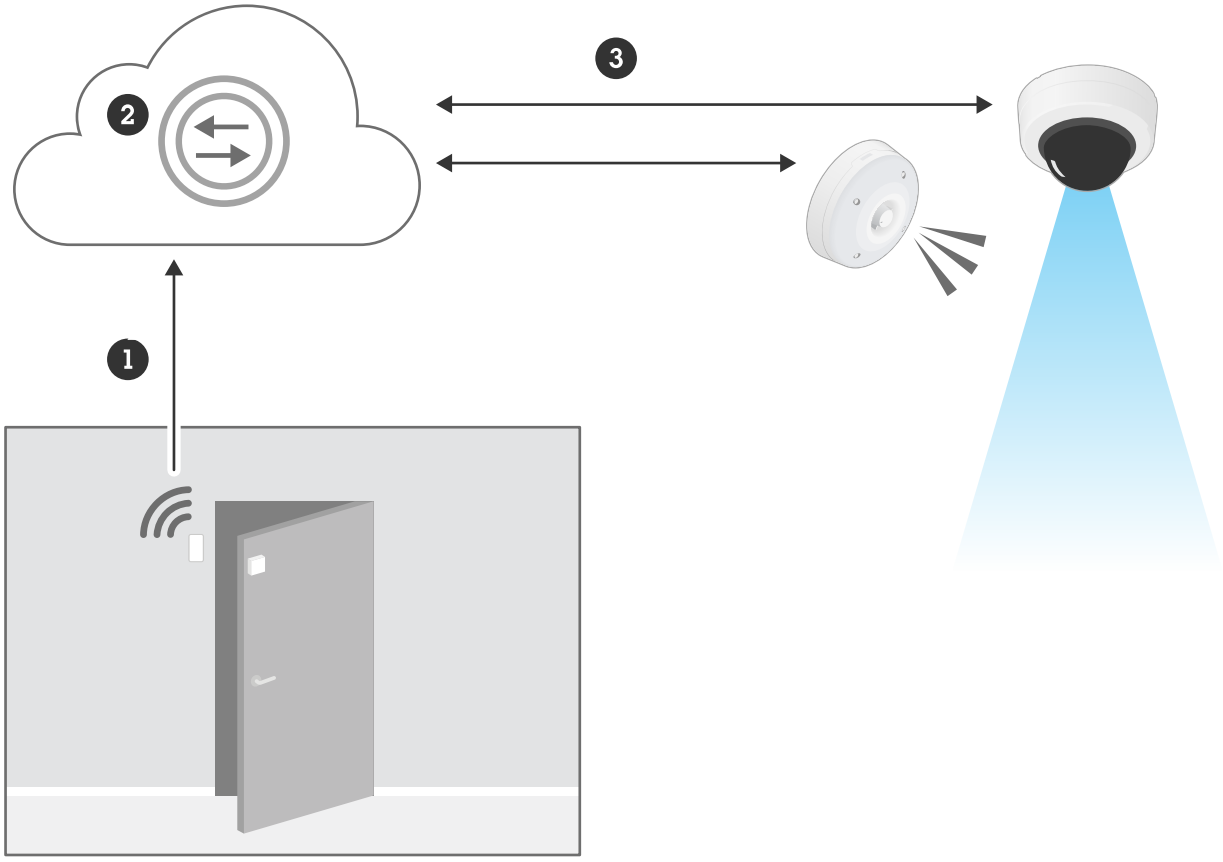


1 *Publish*
2 *Broker MQTT*
3 *Subscribe*

10.2 I dati del sensore per porta su MQTT attivano l'allarme del dispositivo di segnalazione e la registrazione sulla telecamera

Un sensore per porta MQTT di terze parti viene utilizzato per attivare una notifica di evento se la porta viene aperta. Il sensore per porta pubblica un messaggio MQTT al broker MQTT nel cloud. Il dispositivo di

segnalazione e la telecamera si iscrivono all'argomento del sensore per porta e riproducono un allarme se la porta viene aperta, avviando anche una registrazione.



1 *Publish*
2 *Broker MQTT*
3 *Subscribe*

11 Glossario

ACAP	*AXIS Camera Application Platform*: framework per applicazioni che aggiungono funzionalità e intelligenza in modalità edge
Aedes	Broker MQTT
API	*Interfaccia per la programmazione di applicazioni*: codice che consente a due programmi software di comunicare tra loro
AWS™	Piattaforma di servizi cloud
AXIS OS	Sistema operativo dei dispositivi edge Axis
CloudMQTT	Broker MQTT
Eclipse Mosquitto™	Broker di messaggistica open source che implementa i protocolli MQTT
Google Cloud Platform™	Piattaforma di servizi cloud
HiveMQ™	Broker MQTT
HTTP	*Hypertext Transfer Protocol*: protocollo di trasferimento dati utilizzato nel World Wide Web
IBM Cloud®	Piattaforma di servizi cloud
IoT	*Internet of Things*: interconnessione tramite Internet di dispositivi informatici integrati su dispositivi ed elettrodomestici di uso quotidiano
JSON	*JavaScript Object Notation*: formato di file compatto e formato di scambio dati
Microsoft® Azure® IoT	Piattaforma di servizi cloud
Microsoft® Power BI®	Programma software interattivo per la visualizzazione dei dati incentrato sulla business intelligence
MQTT	*Message Queuing Telemetry Transport*: protocollo di messaggistica standard per l'Internet of Things.
Node.js®	Piattaforma di sviluppo open source per l'esecuzione di codice JavaScript lato server
Node-RED®	Strumento di programmazione per il cablaggio dell'Internet of Things
ONVIF®	Organizzazione aperta del settore che fornisce e promuove interfacce standardizzate, per un'interoperabilità efficace dei dispositivi di sicurezza fisica basati su IP.
PHP	Linguaggio di scripting generico orientato allo sviluppo web
Python®	Linguaggio di programmazione generico
RTSP	*Real-Time Streaming Protocol*: protocollo di rete per stabilire e controllare sessioni multimediali tra gli endpoint
TCP	*Transmission Control Protocol*: protocollo di trasporto dati. È uno dei principali protocolli Internet
TLS	*Transport Layer Security*: protocollo che offre riservatezza e integrità delle comunicazioni su reti informatiche
VAPIX®	API (Application Programming Interface) aperta dei prodotti Axis

WebSocket	Protocollo di comunicazione che fornisce canali per la comunicazione bidirezionale su una singola connessione TCP
VMS	*Software di gestione video* o *sistema di gestione video*

12 Marchi

Android e Google Cloud Platform sono marchi di Google LLC.

AWS è un marchio di Amazon.com, Inc. o delle sue affiliate negli Stati Uniti e/o in altri paesi.

Eclipse Mosquitto è un marchio di Eclipse Foundation, Inc.

HiveMQ è un marchio di HiveMQ GmbH.

IBM e IBM Cloud sono marchi di International Business Machines Corporation registrati in molte giurisdizioni in tutto il mondo.

IOS è un marchio o un marchio registrato di Cisco Systems, Inc e/o delle sue affiliate negli Stati Uniti e in altri paesi ed è utilizzato su licenza da Apple, Inc.

JavaScript è un marchio registrato di Oracle Corporation negli Stati Uniti.

Linux è un marchio registrato di Linus Torvalds negli Stati Uniti e in altri paesi.

Microsoft, Windows, Microsoft Azure IoT e Microsoft Power BI sono marchi registrati di Microsoft Corporation.

Node.js e Node-RED sono marchi registrati della OpenJS Foundation negli Stati Uniti e/o in altri paesi.

ONVIF è un marchio di Onvif, Inc.

Python è un marchio registrato della Python Software Foundation.

T10162730/IT/M1.17/2203

Informazioni su Axis Communications

Axis consente un mondo più intelligente e più sicuro creando soluzioni per migliorare la sicurezza e le prestazioni aziendali. Come società di tecnologie di rete e leader nel settore, Axis offre soluzioni nella videosorveglianza, controllo degli accessi, interfono e sistemi audio. Queste sono ottimizzate da applicazioni di analisi intelligente e supportate da formazione di alta qualità.

Axis ha circa 4.000 impiegati dedicati in più di 50 paesi e collabora con partner di tecnologia e integrazione di sistema in tutto il mondo per offrire soluzioni di clienti. Fondata nel 1984, Axis e con sede a Lund, in Svezia