

di Veronica Merenda

IL PILASTRO DELL'IOT



Scalabile, aperto e affidabile, MQTT è nato come protocollo per ridurre al minimo l'utilizzo della banda nelle comunicazioni tra dispositivi

MQTT (Message Queue Telemetry Transport) è un protocollo ISO standard (con architettura client/server), progettato da Arlen Nipper (cirrus Link) e Andy Stanford-Clark (IBM) alla fine degli anni '90 per consentire la comunicazione tra device in scenari di elevata latenza e limitata larghezza di banda.

Oggi questo protocollo è stato riconosciuto dalla OASIS (Organization for the Advancement of Structured Information Standards) come lo standard di riferimento nella comunicazione IoT, apprezzato soprattutto per le caratteristiche di efficienza (riduzione al minimo delle risorse richieste dal client e la larghezza di banda della rete), indipendenza dal contenuto del pacchetto trasportato (payload), bidirezionalità della comunicazione tra dispositivi e server (con possibilità di trasmettere messaggi broadcast a gruppi di oggetti), sicurezza (per la cifratura dei dati e l'autenticazione del client) e scalabilità (per la possibilità di connessione di milioni di dispositivi IoT).

In termini di Quality of Service (insieme di indicatori che stabiliscono il livello di affidabilità e prestazioni del servizio) MQTT fornisce all'utente la possibilità di scegliere tra 3 livelli di servizio (trasporto messaggi). Il primo livello, definito "at most once" (al massimo una volta), provvede alla distribuzione dei messaggi al meglio delle possibilità della rete TCP/IP sottostante, dunque senza garanzia dell'avvenuta ricezione/spedizione del messaggio. Il secondo livello di servizio "at least once" (almeno una volta) assicura il recapito dei messaggi, seppure non possa evitare invii o ricezioni multiple. L'ultimo livello, "exactly once" (esattamente una volta), assicura che i messaggi vengano inviati o ricevuti esattamente una volta.

Funzionamento

A differenza del protocollo HTTP (l'altro più diffuso protocollo di comunicazione basato su TCP/IP), che si basa su un modello request/response, MQTT utilizza lo schema publish/subscribe, che fornisce un'alternativa alla tradizionale architettura client/server. Il modello pub/sub, infatti, disaccoppia il client che invia un messaggio (il publisher) dal client o dai client che ricevono i messaggi (i subscribers). La connessione tra loro è gestita da un terzo componente, il broker MQTT,

che ha il compito di filtrare i messaggi in arrivo e distribuirli correttamente ai subscribers. I publisher e i subscriber non si conoscono e non stabiliscono mai una connessione diretta. Più in dettaglio, i mittenti (publisher) pubblicano i messaggi su un determinato argomento (Topic), sul message broker, mentre ogni destinatario (subscriber) si iscrive ai Topic, del broker, che lo interessano, in questo modo, ogni volta che un nuovo messaggio viene pubblicato su un determinato Topic, il message broker lo distribuisce a tutti i destinatari interessati. Di fatto viene implementata una comunicazione asincrona tra oggetti, integrando diverse sorgenti. È grazie a questo modello che risulta possibile alleggerire di molto la banda e supportare sia i casi d'uso in cui i client sono sempre connessi ("always-connected") sia quelli in cui, a causa dell'inaffidabilità della rete sottostante, gli stessi possono disconnettersi e riconnettersi continuamente ("sometimes-connected").

Applicazioni

MQTT è principalmente utilizzato per la messaggistica e lo scambio di dati tra dispositivi IoT e IIoT industriali (IIoT), come sistemi embedded, sensori, PLC e altri dispositivi industriali. Gli scenari applicativi riguardano principalmente la smart industry e le smart city, dal controllo dello stato della produzione real-time ai sistemi di telemanutenzione, telediagnosi e controllo da remoto dei macchinari, al controllo dei consumi energetici, alla gestione del traffico e smart parking fino alla gestione dinamica di scale mobili, ascensori e display in grandi infrastrutture.

*Veronica Merenda

Technical Support & IIoT Application Engineer

www.linkedin.com/in/veronica-merenda-8921a81a/

