

# MQTT

## MANUAL DE USO

Traducción de las instrucciones originales

Versión: 1.0

Fecha: 07/08/2024

## Índice

1.	Configuración MQTT .....	4
	Parámetros generales - Configuración MQTT .....	4
	Función MQTT .....	7

VERSIÓN	FECHA	CAMBIOS
1,0	07/08/2024	-



Eelectron S.p.A.  
Via Claudio Monteverdi 6, I-20025 Legnano (MI), Italia  
Tel +39 0331.500802 [info@eelectron.com](mailto:info@eelectron.com)



## 1. Configuración MQTT

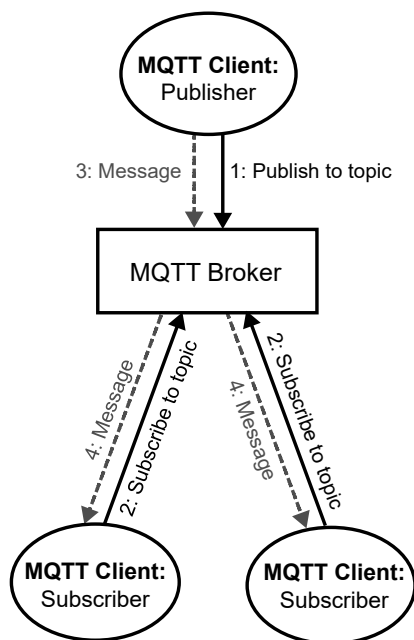
MQTT es un **protocolo de mensajería** diseñado para la transferencia de mensajes y basado en un modelo publish-subscribe.

MQTT se utiliza a menudo en los ambientes de Internet de las Cosas (IoT) y Máquina a Máquina (M2M) para permitir que los dispositivos se comuniquen entre sí y con servidores back-end.

MQTT es útil para permitir una comunicación eficiente y fiable entre dispositivos, sobre todo en ambientes donde los recursos de los dispositivos son limitados.

Con MQTT, un publisher (editor) publica mensajes sobre un tema y un subscriber (abonado) debe inscribirse a esa discusión para ver el mensaje.

MQTT requiere el uso de un broker central y su instrumento, como se muestra en el siguiente diagrama:



### Características principales:

- MQTT utiliza **TCP/IP** para conectarse al broker.
- Los clientes MQTT publican a intervalos regulares un **mensaje keepalive** que indican al broker que el cliente sigue conectado.
- Los clientes **no tienen direcciones** como en los sistemas de correo electrónico, y los mensajes no se envían a los clientes.
- Los mensajes se **publican en un broker con referencia a un tema determinado**.
- La tarea de un broker MQTT es **filtrar los mensajes** según el tema, para luego **distribuirlos a los suscriptores**.
- Un cliente puede recibir estos mensajes suscribiéndose al debate en curso sobre el mismo broker
- No existe una conexión directa entre publisher y subscriber.
- Todos los clientes pueden publicar (transmitir) y suscribir (recibir).
- Los brokers MQTT no suelen almacenar mensajes.

### MQTT puede usarse para:

- Transmitir datos entre dispositivos KNX o IoT a través de servidores back-end, por ejemplo, para recoger datos de sensores o sistemas de control de actuadores.
- Activar acciones en tiempo real, por ejemplo, notificar a una aplicación cuando un sensor detecta un cambio en el estado del ambiente.

biente.

- Soportar la comunicación bidireccional entre dispositivos, por ejemplo, para permitir que una aplicación envíe órdenes a un dispositivo KNX o IoT.
- Gestionar la seguridad de los datos mediante el uso de autenticación y cifrado.

### Parámetros generales - Configuración MQTT

Objeto de comunicación involucrado:

"<General > Conexión al broker de alarma"	1 bit	CRT
---	-------	-----

PARÁMETRO KNX	CONFIGURACIONES
<b>Conexión</b>	
<b>ID client</b>	ee%S
Un <b>client</b> es cualquier dispositivo que ejecute una biblioteca MQTT y se conecte a un broker MQTT en una red. El <b>ID client</b> es un identificador único utilizado en MQTT para identificar a cada cliente que se conecta al broker. El ID de cliente es utilizado por el broker para identificar los mensajes enviados por cada cliente y enviar los mensajes a los clientes apropiados. El ID de cliente debe ser único para cada cliente que se conecte al broker, de lo contrario el broker puede confundir mensajes enviados por diferentes clientes con el mismo ID.	
<b>Dirección del broker</b>	64 byte permitidos
Un broker MQTT es un servidor que recibe todos los mensajes de los clientes MQTT y los enruta a los clientes de destino apropiados.  Este parámetro identifica la dirección del broker MQTT al que conectarse.	
<b>Puerto del broker</b>	1 . 65535
Es el puerto del broker utilizado para la conexión (el valor por defecto es 8883).	
<b>Intervalo de conexión del broker [s]</b>	0 . 255
Define el intervalo de tiempo durante el cual el dispositivo intenta conectarse al broker. En caso de avería, espera el tiempo necesario antes de volver a conectarse.	
<b>Uso de TLS</b>	no / sí
<b>TLS (Transport Layer Security)</b> es un protocolo de seguridad usado para proteger las comunicaciones en red. TLS funciona añadiendo una nivel de criptografía a las comunicaciones de red, de modo que sólo el emisor y el receptor pueden leer el contenido de las comunicaciones. De este modo, TLS protege las comunicaciones de la interceptación o modificación por parte de terceros.	

<b>Validación del certificado</b>	no / sí
<p>La opción "<b>Validación certificado</b>" para TLS (Transport Layer Security) sirve para comprobar que el certificado de seguridad utilizado por el servidor al que se conecta es válido y fiable.</p> <p>Cuando se utiliza TLS para establecer una conexión segura con un servidor, éste envía su certificado de seguridad al cliente. El certificado de seguridad incluye información sobre la identidad del servidor, como el nombre del dominio y el nombre de la organización que lo gestiona, así como una clave criptográfica pública utilizada para proteger las comunicaciones.</p> <p>La opción "Validación del certificado" permite verificar que el certificado de seguridad enviado por el servidor es válido y no ha sido alterado o modificado de ninguna manera.</p> <p>Considere que en algunos casos o estados (depuración de la comunicación) esta opción ayuda al configurador a realizar pruebas de comunicación manteniendo un canal cifrado.</p> <p>Se recomienda activar esta opción al final de la configuración del dispositivo.</p> <p>La carga de los certificados se gestiona desde el "<b>Eelectron Certificate Loader</b>" propiedad de eelectron. Consulte el manual de uso específico del software "<a href="#">Eelectron Certificate Loader</a>".</p>	
<b>Nombre usuario</b>	
Es el nombre de usuario para la autenticación y el acceso al broker.	
<b>Contraseña</b>	
Es la contraseña del usuario para acceder al broker.	
<b>Intervalo keepalive [s] (0 = deshabilitado)</b>	1 . 65535
El intervalo keepalive indica el tiempo utilizado para gestionar la conexión entre el cliente y el broker. El intervalo keepalive lo establece el cliente cuando se conecta al broker y se utiliza para verificar que la conexión sigue activa.	
<b>Sesión de limpieza</b>	no / sí
<p>La <b>sesión de limpieza es una bandera</b> que indica si el cliente debe conservar o eliminar los mensajes y suscripciones no entregados durante la desconexión.</p> <ul style="list-style-type: none"> <li>• Cuando la sesión de limpieza está configurada en <b>sí</b>, el cliente elimina todos los mensajes y suscripciones no entregados cuando se desconecta del broker. De este modo, el cliente no recibe los mensajes enviados durante la desconexión y debe registrarse de nuevo en el broker cuando se vuelva a conectar.</li> <li>• Cuando la sesión de limpieza está configurada en <b>no</b>, el cliente conserva los mensajes y suscripciones no entregados durante la desconexión y los recibe cuando vuelve a conectarse al broker. De este modo, el cliente puede recibir todos los mensajes enviados durante la desconexión sin tener que reintentar.</li> </ul> <p>El valor de la sesión de limpieza depende de las necesidades de la aplicación y debe elegirse adecuadamente para garantizar una buena gestión de la conexión y de los mensajes.</p>	
<b>Gestión de los mensajes Will</b>	
<p>El "<b>mensaje Will</b>", conocido también como "<b>mensaje LWT</b>" (Last Will y Testament message, últimas voluntades y testamento), es una funcionalidad del protocolo MQTT que permite a los clientes MQTT especificar un mensaje a enviar automáticamente a otros clientes MQTT o los servidores MQTT en caso de que el cliente se desconecte de forma anormal, sin enviar un mensaje de desconexión.</p> <p>El "mensaje will" puede utilizarse para varios fines, por ejemplo:</p> <ul style="list-style-type: none"> <li>• Notificar a otros dispositivos o aplicaciones que un dispositivo se ha desconectado de forma anormal, para que otros dispositivos puedan actuar en consecuencia.</li> <li>• Actualizar el servidor MQTT sobre el estado del cliente, para que el servidor pueda cambiar su estado y responder en consecuencia.</li> <li>• Notificar a los operadores del sistema un problema con el dispositivo o la aplicación.</li> </ul>	

<b>Mensaje Will - argumento</b>	ee/%D/%S/estado
<p>En el protocolo MQTT, el nombre del argumento es una cadena simple estructurada jerárquicamente en niveles que el broker utiliza para filtrar los mensajes de cada cliente conectado. Cada nivel de tema está separado por una barra invertida (separador de nivel de tema).</p> <div style="text-align: center;"> </div> <p>Este parámetro define el argumento del mensaje will.</p>	
<b>Mensaje Will - carga útil</b>	30 byte permitidos
Define el contenido del mensaje.	
<b>Mensaje Will - conservación</b>	no / sí
<p>Si está configurado en <b>sí</b>, el broker almacena el último mensaje almacenado y la QoS correspondiente para el tema. Todo cliente que se suscriba a un modelo de temas que corresponda al tema del mensaje almacenado recibe el mensaje almacenado inmediatamente después de la suscripción. El broker sólo almacena un mensaje almacenado por tema.</p> <p>Si está configurado en <b>no</b>, el mensaje no se reenvía.</p>	
<b>Mensaje Will - Calidad del servicio (QoS)</b>	0 / 1 / 2
<p><b>QoS (Calidad del servicio)</b> es un atributo asignado a un mensaje MQTT individual, es un acuerdo entre emisor y receptor que define cómo se entrega y transmite un mensaje. Permite a los clientes tener en cuenta la fiabilidad de la red. El intermediario y el cliente pueden retransmitir mensajes y garantizar su entrega, facilitando la comunicación en redes poco fiables.</p> <ul style="list-style-type: none"> <li>• <b>(0) Un mensaje se entrega como máximo una vez.</b> Puede que no se entregue en absoluto.</li> <li>• <b>(1) El mensaje se entrega al menos una vez.</b> Debe confirmarse la recepción del mensaje. Si no se recibe acuse de recibo, el mensaje se volverá a enviar. Este proceso se repite hasta que se confirma la recepción del mensaje y puede dar lugar a que el mismo mensaje se envíe y procese varias veces.</li> <li>• <b>(2) Los mensajes enviados siempre se entregan exactamente una vez.</b> Es el modo de transferencia más lento y fiable en una red MQTT. Antes de que el emisor borre un mensaje, la transmisión entre emisor y receptor se repite al menos dos veces.</li> </ul> <p>Para más información consulte: <a href="https://mqtt.org/">https://mqtt.org/</a></p>	
<b>Gestión de los mensajes</b>	
<p>El "<b>Mensaje Birth</b>" es un mensaje que el cliente MQTT envía automáticamente al broker MQTT cuando el cliente se conecta con éxito al broker. A diferencia del "mensaje will", el "mensaje birth" se envía una sola vez, cuando el cliente se conecta al broker MQTT.</p> <p>El "mensaje birth" puede usarse para varios fines, por ejemplo:</p> <ul style="list-style-type: none"> <li>• Notificar al broker MQTT o a otros clientes MQTT que el cliente acaba de conectarse y está listo para recibir mensajes.</li> <li>• Actualizar el broker MQTT u otros clientes MQTT sobre el estado del cliente recién conectado.</li> <li>• Enviar información sobre el cliente, incluyendo nombre, ID u otra información de configuración.</li> </ul>	
<b>Mensaje Birth - argumento</b>	ee/%D/%S/estado
Este parámetro define el argumento del mensaje birth.	
<b>Mensaje Birth - carga útil</b>	30 byte permitidos
Define el contenido del mensaje.	
<b>Mensaje Birth - conservación</b>	no / sí
Véase " <b>Will - retain</b> "	
<b>Mensaje Birth - Calidad del servicio (QoS)</b>	0 / 1 / 2

Véase “Mensaje Will - Calidad del servicio (QoS)”	
<b>Argumento general</b>	
<b>Prefijo</b>	ee/%D/%S
El prefijo del argumento se utiliza para crear una jerarquía de subtemas, por ejemplo, para enviar mensajes a dispositivos de una determinada zona geográfica o a dispositivos pertenecientes a una determinada categoría. Por ejemplo, el prefijo "casa/habitación 1/" sólo puede utilizarse para enviar mensajes a los dispositivos de la habitación 1 de una casa.	
<b>Regla de suscripción</b>	
Añade un nivel al argumento, que depende de la regla de definición del parámetro de posición, que identifica si el mensaje es un comando asociado (es decir, recibido por el broker y enviado al bus).	
<b>Regla de publicación</b>	
Añade un nivel al argumento, que depende de la regla de definición del parámetro de posición, que identifica si el mensaje es un mensaje de estado (recibido del bus y enviado al broker).	
<b>Regla de posición</b>	fin del argumento / fin del prefijo
Es la posición en la que debe añadirse el nivel que identifica el tipo de mensaje.	
<b>Retraso mínimo entre mensajes objeto MQTT [ms]</b>	30 / 40 / 50 / 75 / 100 / 150 200 / 250
Define el intervalo de tiempo mínimo que transcurre entre dos mensajes MQTT.	
<b>Modelo cadena x (x = 1... 8)</b>	16 byte permitidos
Permite asociar una cadena a un modelo x (1...8). La cadena se puede llamar en los argumentos introduciendo "%x".	
<b>Ejemplo</b> Modelo 1 = casa Argumento = ee/%D/%S/%1/cmd Resultado = ee/bridge/006c12345678/home/cmd	
<b>Configuración alarma</b>	
<b>Alarma de conexión al broker</b>	deshabilitado / habilitado
Habilita el objeto "<General > Alarma de conexión al broker" y notifica si la conexión con el broker está activa.	
<b>Telegrama de alarma</b>	telegrama "0" / telegrama "1"
Define el telegrama enviado al objeto "<General > Alarma de conexión al broker" cuando la alarma está deshabilitada.	

## Función MQTT

Objeto de comunicación involucrado:

"<MQTT x   > Objeto"	1 bit... 4 byte	CRT/CW/CRWT
----------------------	-----------------	-------------

PARÁMETRO KNX	CONFIGURACIONES
<b>Función MQTT</b>	ninguno subscribe (de broker a BUS) publica (de BUS a broker) subscribe/publica
Con este parámetro es posible configurar el modo de uso del protocolo MQTT:	
<b>ninguno:</b> no usar <b>subscribe (del broker al BUS):</b> el dispositivo se suscribe al argumento y envía la carga útil de los mensajes recibidos del broker en formato de cadena al bus, de acuerdo con el DPT configurado. Flag del objeto: CRT <b>publish (de BUS a broker):</b> el dispositivo publica el argumento en el broker y crea la carga útil transformando el valor del objeto en una cadena de caracteres. Flag del objeto: CW	
<b>Ejemplo de publicación</b>	
Prefijo = ee/%D/%S Rol de publicación = /sts Posición de rol = fin del argumento Argumento Mqtt = /1 Valor = 1  El argumento final será: ee/%D/%S/1/sts con carga útil 1  con posición de rol = fin del prefijo  El argumento final será: ee/%D/%S/sts/1 con carga útil 1	
<b>Subscribe / publish:</b> Crear automáticamente dos argumentos las reglas descritas anteriormente para las funciones subscribe (suscripción) y publish (publicación) Flag del objeto: CRWT	
<b>Nombre MQTT</b>	16 byte permitidos
Este parámetro define el nombre del módulo MQTT; el nombre puede utilizarse para identificar rápidamente la funcionalidad.	
<b>Argumento MQTT</b>	30 byte permitidos
Identifica el argumento. Consulte: " <a href="#">Eelectron Certificate Loader</a> " en la página 5.	
<b>Dimensión DPT</b>	
Este parámetro define el DPT de la función MQTT. El telegrama puede ser: <ul style="list-style-type: none"> <li>• 1 bit</li> <li>• 4 bit</li> <li>• 1 byte (señalado, no señalado - con signo, sin signo)</li> <li>• 2 byte (señalado, no señalado, flotamiento - con signo, sin signo, variable)</li> <li>• 4 byte (señalado, no señalado, flotamiento - con signo, sin signo, variable)</li> </ul>	
<b>Tipo DPT</b>	0 - 255 0 - 100% ángulo Modalidad HVAC
En el caso de 1 byte no señalado, especifica el DPT del objeto.	
<b>Conservación</b>	no / sí
Véase “Will - retain”.	

Calidad del servicio (QoS)	0 / 1 / 2
<p><b>QoS (Calidad del servicio)</b> es un atributo asignado a un mensaje MQTT individual, es un acuerdo entre emisor y receptor que define cómo se entrega y transmite un mensaje. Permite a los clientes tener en cuenta la fiabilidad de la red. El intermediario y el cliente pueden retransmitir mensajes y garantizar su entrega, facilitando la comunicación en redes poco fiables.</p> <ul style="list-style-type: none"> <li>• <b>(0) Un mensaje se entrega como máximo una vez.</b> Puede que no se entregue en absoluto.</li> <li>• <b>(1) El mensaje se entrega al menos una vez.</b> Debe confirmarse la recepción del mensaje. Si no se recibe acuse de recibo, el mensaje se volverá a enviar. Este proceso se repite hasta que se confirma la recepción del mensaje y puede dar lugar a que el mismo mensaje se envíe y procese varias veces.</li> <li>• <b>(2) Los mensajes enviados siempre se entregan exactamente una vez.</b> Es el modo de transferencia más lento y fiable en una red MQTT. Antes de que el emisor borre un mensaje, la transmisión entre emisor y receptor se repite al menos dos veces.</li> </ul> <p>Para más información consulte: <a href="https://mqtt.org/">https://mqtt.org/</a></p>	

### Ejemplo

Consideramos el siguiente modelo de datos:

- comunicación de los datos de temperatura del BUS KNX al cliente MQTT.
- comunicación del punto de ajuste de temperatura del cliente MQTT al BUS KNX.
- 1 bit de comunicación del mando on/off de la luz.
- comunicación de un mando porcentual para el control de las persianas

Paso 1: Configuración ETS

[https://download.eelectron.com/ETS%20-%20configuration\\_eg.mp4](https://download.eelectron.com/ETS%20-%20configuration_eg.mp4)

Paso 2: Configuración cliente MQTT

[https://download.eelectron.com/MQTT%20Explorer\\_eg.mp4](https://download.eelectron.com/MQTT%20Explorer_eg.mp4)



El valor enviado del cliente MQTT al BUS KNX es el valor absoluto del objeto de comunicación, por ej. mando % 100% = 100 , mando RGB FFF = FFF)

Sólo en caso de DPT de 4 bit, el valor a escribir en el cliente MQTT debe tener esta información:

[0 - 1] ; [0 - 7], donde 0 indica una disminución, y 1 y 7 indican un aumento:

0	break	
1	1	100.0%
2	2	50.0%
3	4	25.0%
4	8	12.5%
5	16	6.3%
6	32	3.1%
7	64	1.6%



Para una análisis detallado de los datos, existen plataformas específicas donde es posible registrar el dispositivo.