MANUAL DE ARQUITECTURA DE TÓPICOS, CALIDAD DE SERVICIO (QoS) Y SEGURIDAD MQTT

Diseño de Espacios de Nombres ISA-95, Dimensionamiento de Tráfico, Retención, LWT y Endurecimiento de Mosquitto

---

1. Paradigma Publicador/Suscriptor frente a Sondeo Maestro-Esclavo

En el nivel de campo (**Tecnología de Operación — OT**), protocolos como **Modbus RTU sobre RS-485** utilizan un esquema *Maestro-Esclavo por sondeo secuencial (*polling*) **altamente determinista, pero con acoplamiento directo y una capacidad limitada a un único maestro físico en el bus. Cuando múltiples sistemas de nivel superior (**SCADA, Historiador de Base de Datos, Analítica en la Nube y Paneles Móviles**) necesitan consumir las variables del proceso simultáneamente, interrogar de forma directa el bus serial resulta inviable.

El protocolo *MQTT (*Message Queuing Telemetry Transport*, estándar OASIS / ISO/IEC 20922) **resuelve esta frontera mediante un** Broker centralizado **que desacopla a los productores de datos (**Publicadores**) de los consumidores (**Suscriptores**) en tres dimensiones fundamentales:

- **Desacoplamiento de Espacio:** El nodo de borde (Raspberry Pi) no necesita conocer las direcciones IP ni la cantidad de clientes SCADA o bases de datos existentes; únicamente publica en la IP del Broker.
- **Desacoplamiento de Tiempo:** Gracias a las sesiones persistentes (`clean_session=False` / `Session Expiry Interval`) y a los **mensajes retenidos (`retain=True`)**, un suscriptor puede conectarse horas después de la emisión y recibir instantáneamente el último estado conocido.
- **Desacoplamiento de Sincronización:** Las publicaciones y recepciones operan de manera asíncrona mediante eventos sin bloquear el bucle de adquisición de control.

---

2. Modelado Matemático de Sobrecarga y Caudal de Red en MQTT

Una trama MQTT posee una cabecera fija ultracompacta de tan solo $2\ \text{bytes}$ (1 byte de tipo de control y banderas + 1 a 4 bytes de longitud restante codificada en base 128). Para un mensaje de publicación (`PUBLISH`), el tamaño total en bytes de aplicación L_{MQTT} viene dado por:

$$L_{MQTT} = L_{fija} + \leq ft(2 + L_{tópico} \right) + L_{PacketID} + L_{payload}$$

Donde:

- $L_{fija} \in [2,\ 5]\ \text{bytes}$ es la cabecera fija.
- $2 + L_{tópico}$ corresponde a los 2 bytes de longitud UTF-8 más la cadena del tópico (por ejemplo, `invernadero/zona_norte/nodo_01/estado/resumen` tiene $L_{tópico} = 45\ \text{bytes}$).
- $L_{PacketID} = 0\ \text{bytes}$ para **QoS 0**, y $L_{PacketID} = 2\ \text{bytes}$ para **QoS 1** y **QoS 2**.
- $L_{payload}$ es la longitud útil del cuerpo JSON (por ejemplo, $160\ \text{bytes}$).

Si el sistema cuenta con N nodos publicando cada T_{pub} segundos hacia M suscriptores con un nivel de calidad de servicio $q \in \{0,\ 1,\ 2\}$, el número total de paquetes MQTT intercambiados por ciclo de publicación es:

$$N_{paquetes}(q,\ M) = 2^q \cdot (1 + M)$$

- **En QoS 0 ($q=0$):** $1\ \text{paquete}$ del emisor al broker (`PUBLISH`) + $M\ \text{paquetes}$ del broker a los suscriptores $= 1 + M$.

- **En QoS 1 ($q=1$):** 2 paquetes por salto (`PUBLISH` + `PUBACK`) $= 2 \cdot (1 + M)$.
- **En QoS 2 ($q=2$):** 4 paquetes por salto (`PUBLISH` → `PUBREC` → `PUBREL` → `PUBCOMP`) $= 4 \cdot (1 + M)$.

---

3. Reglas de Ingeniería para Jerarquía de Tópicos (Alineación ISA-95)

- **Estructura de General a Específico:**
  `sitio/area/unidad_borde/categoria/variable`
  Ejemplo: `invernadero/zona_norte/nodo_01/sensores/temperatura`
- **Prohibición de Barra Inicial (`/`):** Un tópico que inicia con `/invernadero/...` introduce un nivel vacío `""` innecesario en el árbol de enrutamiento del broker.
- **Caracteres ASCII Minúsculos y Guion Bajo (`snake_case`):** Evite espacios en blanco, tildes o caracteres especiales dentro del nombre del tópico para garantizar compatibilidad total con puentes SQL, InfluxDB y Kafka.
- **Uso Correcto de Comodines (*Wildcards — *Exclusivos para `SUBSCRIBE`):**
- **Comodín de Un Nivel (`+`):** Sustituye exactamente un nivel entre barras. Ejemplo: `invernadero/+/+/sensores/temperatura` suscribe la temperatura de todos los nodos en todas las zonas.
- **Comodín Multinivel (`#`):** Sustituye cero o más niveles y **debe ubicarse obligatoriamente al final del tópico**. Ejemplo: `invernadero/zona_norte/nodo_01/#` .

---

4. Semántica de Mensajes Retenidos (`Retain`) y Testamento (`LWT`)

Mecanismo	Configuración Recomendada	Propósito Operativo Industrial
Telemetría Periódica (`.../estado/resumen`)	`QoS = 1` , `Retain = False`	Evita que un cliente reciba una lectura histórica como si acabara de ocurrir en el milisegundo actual.
Parámetros y Límites (`.../configuracion/limites`)	`QoS = 1` , `Retain = True`	Permite que cualquier panel HMI recién abierto conozca de inmediato la consigna vigente sin esperar cambios.
Estado de Salud + LWT (`.../estado/conexion`)	`QoS = 1` , `Retain = True`	Combina el **Birth Message** (`"CONECTADO"`) con el **LWT** (`"DESCONECTADO_INESPERADO"`) almacenado en el Broker.
Comandos de Actuación (`.../actuadores/comando`)	`QoS = 1` , `Retain = False`	**Crítico:** Jamás usar `Retain = True` en tópicos de comando directo para evitar re-ejecuciones accidentales al reconectar el puente.

El tiempo máximo de detección de caída abrupta por parte del Broker Mosquitto obedece la especificación OASIS MQTT:

> $$T_{\text{detección LWT}} = 1.5 \times T_{\text{keepalive}}$$

Para $T_{\text{keepalive}} = 60\ \text{s}$, si el puente sufre un corte eléctrico o pérdida de enlace físico, el Broker espera exactamente $90\ \text{s}$ sin recibir paquetes de control ni `PINGREQ` antes de despachar el mensaje LWT a todos los clientes supervisores.