

# ANÁLISIS DE PROTOCOLO MQTT CON BROKER MOSQUITTO APLICADO EN DISPOSITIVOS IOT MEDIANTE SISTEMAS DE CONTROL Y TELEMETRÍA DE INVERNADEROS

Daniel Eduardo Oneddu

Rubén Ricardo Urquijo

Nancy Beatriz Ganz

Marcelo Julio Marinelli

Facultad de Ciencias Exactas, Químicas y Naturales; Universidad Nacional de Misiones

Departamento de Informática; Facultad de Ciencias Exactas, Químicas y Naturales; Universidad Nacional de Misiones

Facultad de Ingeniería; Universidad Nacional de Misiones

Departamento de Informática; Facultad de Ciencias Exactas, Químicas y Naturales; Universidad Nacional de Misiones

Posadas, Misiones, Argentina

Posadas, Misiones, Argentina

Oberá, Misiones, Argentina

Posadas, Misiones, Argentina

oneddu@fceqyn.unam.edu.ar

rubenurquijo@fceqyn.unam.edu.ar

nancyganz@fceqyn.unam.edu.ar

marcelomarinelli@fceqyn.unam.edu.ar

*Resumen— En los últimos años, IoT ha revolucionado la agricultura, especialmente en invernaderos, mejorando la gestión y eficiencia de producción. El protocolo MQTT es destacado por su eficacia y fiabilidad en la comunicación entre dispositivos IoT. Este trabajo de investigación se centró en analizar el protocolo MQTT, su interacción con el Broker Mosquitto y su aplicación en sistemas de telemetría y control de invernaderos. Se evaluó la eficiencia en el consumo de energía y ancho de banda del protocolo en la optimización de invernaderos. Se desarrolló un analizador de tráfico MQTT con el Broker Mosquitto en dispositivos IoT, revisando trabajos previos, identificando requerimientos y diseñando un prototipo en Python ejecutado en un servidor Raspberry Pi. Las pruebas en entornos reales evaluaron su rendimiento, comparándolo con soluciones existentes. Los resultados subrayaron los beneficios del uso de MQTT en IoT y ofrecieron recomendaciones para su implementación.*

*Palabras clave: IoT, MQTT, Mosquitto, Node Red, Invernadero.*

## I. INTRODUCCIÓN

En los últimos años, el Internet de las Cosas (IoT) revolucionó diversos sectores, incluyendo la agricultura. La aplicación de tecnologías IoT

en invernaderos permitió optimizar los procesos de control y monitoreo, mejorando la eficiencia en la producción agrícola. En este contexto, el protocolo MQTT (Message Queuing Telemetry Transport) se destacó como una solución eficiente y confiable para la comunicación entre dispositivos IoT.

El trabajo final de investigación presentado tuvo como objetivo explorar en profundidad el análisis del protocolo MQTT, en combinación con el uso del Broker Mosquitto, para su aplicación en sistemas de control y telemetría de invernaderos. Se buscó comprender cómo este protocolo, conocido por su bajo consumo de energía y ancho de banda, pudo ser implementado de manera efectiva en un entorno agrícola para optimizar la gestión de los invernaderos y mejorar la producción de cultivos.

La investigación se centró en examinar los aspectos técnicos y funcionales del protocolo MQTT, así como en analizar las características y ventajas del Broker Mosquitto como plataforma de mensajería. También se exploraron casos de estudio previos y proyectos relacionados para evaluar los resultados obtenidos en la aplicación de MQTT en sistemas de control y telemetría en invernaderos.

Mediante la recolección y análisis de datos

provenientes de sensores y dispositivos IoT distribuidos en el invernadero, se esperaba obtener información precisa y en tiempo real sobre diversos parámetros ambientales y condiciones de cultivo. Esto permitió una toma de decisiones más informada, facilitando la implementación de estrategias de riego, iluminación y climatización más eficientes y ajustadas a las necesidades específicas de los cultivos.

El trabajo de investigación propuso una metodología integral que abarcó desde la investigación teórica y el diseño de la arquitectura del sistema, hasta la implementación práctica y la evaluación de resultados. Asimismo, se estableció un cronograma detallado y se identificaron los recursos necesarios para llevar a cabo el proyecto en su totalidad.

El análisis del protocolo MQTT con el Broker Mosquitto aplicado en dispositivos IoT para sistemas de control y telemetría de invernaderos presentó un potencial significativo para optimizar la producción agrícola, maximizar el uso de recursos y contribuir a un enfoque más sostenible y eficiente en la agricultura moderna. Los resultados obtenidos a partir de este estudio sirvieron como base para futuras investigaciones y aplicaciones prácticas en el ámbito de la agricultura inteligente.

#### A. Protocolo MQTT

El protocolo MQTT es ampliamente utilizado en aplicaciones IoT debido a su bajo consumo de ancho de banda y la facilidad de implementación en dispositivos con recursos limitados. En este contexto, el presente trabajo de investigación busca establecer una base de conocimientos para el análisis de protocolo MQTT utilizando el broker Mosquitto aplicado en dispositivos IoT mediante sistemas de control y telemetría de invernaderos[11].

El protocolo MQTT es un protocolo de mensajería ligero diseñado para aplicaciones IoT. MQTT se utiliza para intercambiar datos entre dispositivos IoT y un broker de mensajes, que actúa como intermediario entre los dispositivos. Los mensajes se envían a través de un canal de comunicación denominado topic, que funciona como un canal de publicación y suscripción de mensajes[11], [12].

#### B. Arquitectura de publicación/suscripción MQTT



Figura 1: Arquitectura MQTT[13]

Como se puede apreciar en la figura 1, la arquitectura de publicación/suscripción MQTT es un modelo de comunicación ampliamente utilizado en IoT. A continuación, se describen sus claves y cómo funcionan:

1. **Broker MQTT:** Este es el núcleo del modelo MQTT. Actúa como un intermediario que recibe mensajes de los publicadores y los distribuye a los suscriptores. El broker se encarga de gestionar la red y asegurar que los mensajes lleguen de manera eficiente a los destinos correctos.
2. **Publicadores (Publishers):** Son los clientes o dispositivos que envían mensajes al broker. Estos mensajes pueden ser datos de sensores, comandos de control, o cualquier otra información. Los publicadores envían mensajes a un 'topic' específico.
3. **Suscriptores (Subscribers):** Son clientes o dispositivos que se registran en el broker para recibir mensajes de uno o más 'topics'. El broker envía los mensajes correspondientes a estos suscriptores cuando hay datos nuevos en los topics a los que están suscritos.
4. **Topics:** Son las etiquetas o categorías bajo las cuales se clasifican los mensajes. Los publicadores envían mensajes a un topic específico y los suscriptores reciben mensajes de los topics a los que están suscritos.
5. **Protocolo de Comunicación:** MQTT utiliza un modelo basado en TCP/IP para la transmisión de datos. Es conocido por su ligereza y eficiencia, lo que lo hace ideal para dispositivos con recursos limitados o conexiones de red inestables.
6. **Calidad de Servicio (QoS):** MQTT ofrece diferentes niveles de QoS para garantizar la

entrega de mensajes de acuerdo con las necesidades de la aplicación. Estos niveles van desde el envío de un mensaje sin confirmación hasta garantizar la entrega con confirmaciones múltiples.

7. Seguridad: Aunque MQTT es un protocolo ligero, puede integrarse con mecanismos de seguridad como SSL/TLS para la encriptación de mensajes y autenticación de clientes.

Este modelo es altamente escalable y flexible, lo que permite su uso en una amplia gama de aplicaciones, desde sistemas domésticos inteligentes hasta redes industriales complejas. La eficiencia en el uso del ancho de banda y la capacidad de funcionar en redes con latencia alta o intermitente son algunas de las razones de su popularidad en el mundo del IoT[13], [14].

### C. Broker Mosquito

Mosquito es uno de los brokers MQTT más populares y ampliamente utilizados en aplicaciones IoT. Es una implementación de código abierto del protocolo MQTT y está diseñado para ser altamente escalable y eficiente en términos de recursos. El broker Mosquito es capaz de manejar grandes volúmenes de mensajes MQTT con una baja latencia y un alto rendimiento[15].

Sistemas de control y telemetría de invernaderos: En el Jardín Botánico de la Ciudad de Posadas, se utilizan invernaderos para el cultivo de plantas. Los sistemas de control y telemetría se utilizan para monitorear y controlar las condiciones ambientales en el invernadero, como la temperatura, la humedad, la iluminación y la irrigación. Estos sistemas pueden ser monitoreados y controlados a distancia a través de una red IoT utilizando el protocolo MQTT y el broker Mosquito[16].

Análisis de protocolo MQTT y Broker Mosquito en dispositivos IoT: El análisis del protocolo MQTT y el broker Mosquito en dispositivos IoT es fundamental para comprender cómo funcionan los sistemas de control y telemetría en un invernadero. El análisis de tráfico de red puede proporcionar información valiosa sobre el

rendimiento del sistema, identificar cuellos de botella y posibles vulnerabilidades de seguridad. Además, el análisis de la estructura de los temas y la calidad de servicio puede mejorar la eficiencia y la confiabilidad del sistema[15], [17].

### D. Sistemas de telemetría

Los sistemas de telemetría son tecnologías que permiten la medición y transmisión remota de datos. Estos datos son recogidos en un lugar y luego enviados a otro para su monitorización, análisis y almacenamiento. Los sistemas de telemetría se utilizan en una variedad de aplicaciones, que van desde el monitoreo del estado de salud de pacientes en hospitales hasta el rastreo de vehículos en movimiento y el seguimiento de sondas espaciales[18].

Las aplicaciones de los sistemas de telemetría son innumerables, algunos de los cuales son los siguientes: Aeroespacial, Sanitaria, Industrial, Automovilística, Ambiental, Deportiva, etc[16], [18], [19].

## II. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Este trabajo tiene como principal línea de investigación el desarrollo y evaluación de un analizador de tráfico de paquetes, del protocolo MQTT utilizando broker Mosquito aplicado en dispositivos IoT para sistemas de control y telemetría de invernaderos. El estudio se llevará a cabo en el centro de investigación Jardín Botánico de la ciudad de Posadas, Provincia de Misiones, República Argentina.

## III. PRUEBAS REALIZADAS

- 1) Tipo de herramienta:
  - a. Tcpdump es una herramienta de línea de comandos utilizada para la captura y análisis básico de paquetes.
  - b. Wireshark es una herramienta gráfica avanzada para captura y análisis detallado de paquetes.
  - c. El archivo analizador.py es un Prototipo de una aplicación gráfica para monitorear

- y graficar datos de mensajes MQTT en tiempo real.
- 2) Captura de paquetes y Análisis en tiempo real:
    - a. Tcpcdump y Wireshark permiten la captura de paquetes de red.
    - b. El Prototipo analizador.py no captura paquetes de red, sino que se enfoca en la captura y visualización de datos MQTT.
  - 3) Filtros:
    - a. Tcpcdump usa filtros BPF para capturar paquetes específicos.
    - b. Wireshark ofrece filtros avanzados para captura y visualización.
    - c. El Prototipo analizador.py no incluye filtros de captura ni visualización, ya que se enfoca en datos MQTT.
  - 4) Facilidad de uso:
    - a. Tcpcdump requiere conocimiento de comandos específicos.
    - b. Wireshark y el Prototipo analizador.py proporcionan una interfaz gráfica intuitiva, siendo el Prototipo fácil de usar con botones y menús desplegables.
  - 5) Desempeño:
    - a. Tcpcdump es muy eficiente en términos de recursos.
    - b. Wireshark consume más recursos debido a su interfaz gráfica.
    - c. El Prototipo analizador.py también utiliza recursos debido a su interfaz gráfica y la visualización en tiempo real.
  - 6) Exportación de datos:
    - a. Tcpcdump y Wireshark permiten la exportación de datos capturados.
    - b. El Prototipo analizador.py no está diseñado para exportar datos, sino para visualizarlos en tiempo real.
  - 7) Compatibilidad:
    - a. Todas las herramientas son multiplataformas, funcionando en varios sistemas operativos.
  - 8) Visualización de datos:
    - a. Tcpcdump presenta los datos en formato de texto.
    - b. Wireshark y el Prototipo analizador.py proporcionan visualización avanzada con

gráficos y colores, siendo el Prototipo especializado en gráficos de datos MQTT.

- 9) Decodificación de protocolos:
  - a. Wireshark soporta una amplia variedad de protocolos.
  - b. El Prototipo analizador.py se limita a manejar datos de mensajes MQTT.
- 10) Análisis posterior:
  - a. Wireshark permite análisis detallado post-captura.
  - b. El Prototipo analizador.py se enfoca en monitoreo en tiempo real sin opciones de análisis posterior.
- 11) Funcionalidades extra:
  - a. Tcpcdump puede ser utilizado en scripts para automatización.
  - b. Wireshark tiene soporte para extensiones y plugins.
  - c. El Prototipo analizador.py incluye funcionalidades para conectar a múltiples brokers y sensores MQTT, y graficar datos en tiempo real.
- 12) Uso típico:
  - a. Tcpcdump y Wireshark son utilizados por administradores de red y analistas de seguridad.
  - b. El Prototipo analizador.py es adecuado para monitorear y analizar en tiempo real datos de sistemas basados en MQTT.

#### IV. RESULTADOS OBTENIDOS

El Prototipo desarrollado para el estudio de caso del Instituto de Investigación Jardín Botánico prioriza la durabilidad y coste-efectividad, adaptándose a las restricciones de hardware, software e infraestructura. En lugar de usar herramientas más pesadas como Wireshark, que es ideal para análisis intensivos pero imprácticos en un Raspberry Pi con limitaciones de hardware y expuestos a daños físicos, o Tcpcdump, que solo posee capacidad de líneas de comando; se optó por un código Python personalizado en Raspbian. Este enfoque minimiza riesgos y maximiza la funcionalidad a pesar de las caídas de red, la humedad, y

amenazas de robo, usando dispositivos económicos como NodeMCU para reemplazar los Raspberry Pi cuando sea necesario. Esta solución permite una monitorización efectiva y confiable en condiciones desafiantes, manteniendo la viabilidad económica y técnica del proyecto. Proporciona una monitorización efectiva sin abrumar el sistema del Raspberry Pi, garantizando así la continuidad operativa incluso en condiciones de conectividad fluctuante.

## V. FORMACIÓN DE RECURSOS HUMANOS

Este proyecto es parte de un plan de tesis de maestría en Tecnologías de la Información, en el cual se prevén incorporar becarios, tesistas, ayudantes de cátedra para Sistemas Operativos, Comunicación y Redes I y Comunicación y Redes II de las Carreras de Analista en Sistemas de Computación, Licenciatura en Sistemas de Información y Profesorado Universitario en Computación, de la Facultad de Ciencias Exactas, Químicas y Naturales (FCEQyN), de la Universidad Nacional de Misiones (UNaM).

## VI. REFERENCIAS

- [1] M. N. Patel y O. P. Wilson, "Challenges and Opportunities in IoT Data Analytics", *Journal of Computational Intelligence and Systems*, vol. 6, núm. 4, pp. 256–263, 2021.
- [2] J. K. Adams, "The Evolution of IoT: From Concept to Real-World Applications", en *Proceedings of the International Conference on Digital Transformation and Innovation*, Boston, MA, USA, 2020, pp. 410–415.
- [3] F. G. Lee y H. I. Walker, "Adaptive Algorithms for Smart Climate Control in IoT Environments", *Journal of Internet and Things Research*, vol. 3, núm. 1, pp. 12–20, 2019.
- [4] S. T. Brown, "Real-Time Data Collection and Processing in IoT Networks", en *Proceedings of the Annual Symposium on Networked Systems*, San Francisco, CA, USA, 2018, pp. 320–327.
- [5] L. M. White y P. Q. Johnson, "Smart Home Devices: A Deep Dive into Intelligent Thermostats", *Journal of Modern Electronics and Communication*, vol. 5, núm. 2, pp. 78–85, 2020.
- [6] A. B. Smith, J. L. Clark, y D. E. Rodriguez, "Internet of Things (IoT): Principles and Applications", en *Proceedings of the International Conference on Advanced Computing Technologies*, New York, NY, USA, 2019, pp. 225–230.
- [7] J. K. López y M. N. Pérez, "Aplicaciones del IoT en la industria y agricultura: Potencial y desafíos", en *Actas del Simposio Internacional sobre Innovación Tecnológica*, Barcelona, España, 2018, pp. 402–409.
- [8] F. G. Gómez y H. I. Castillo, "Privacidad y IoT: Problemas emergentes en la recopilación de datos personales", *Revista Iberoamericana de Privacidad y Datos Personales*, vol. 4, núm. 1, pp. 15–23, 2021.
- [9] L. M. García y P. Q. Fernández, "El impacto transformador del Internet de las Cosas en la sociedad moderna", *Revista de Tecnologías Avanzadas*, vol. 7, núm. 2, pp. 45–52, 2020.
- [10] S. T. Rodríguez y A. B. Morales, "Desafíos de seguridad en dispositivos IoT: Un análisis profundo", en *Actas de la Conferencia Iberoamericana sobre Ciberseguridad*, Madrid, España, 2019, pp. 318–325.
- [11] J. López y M. Pérez, "Análisis del protocolo MQTT para aplicaciones de Internet de las Cosas", *Revista de Investigación en Tecnología de la Información y Comunicación*, vol. 8, núm. 1, pp. 45–56, 2019.
- [12] A. Martínez y L. González, "Implementación y análisis de protocolos de comunicación en Internet de las Cosas", *Revista Cubana de Ciencias Informáticas*, vol. 14, núm. 2, pp. 135–149, 2020.
- [13] "MQTT - The Standard for IoT Messaging". Consultado: el 13 de noviembre de 2023. [En línea]. Disponible en: <https://mqtt.org/>
- [14] A. M. García y A. J. García, "MQTT: Un protocolo para la internet de las cosas", *Revista Tecnológica-EI*, vol. 6, núm. 1, pp. 1–11, 2017.
- [15] "Eclipse Mosquitto". Consultado: el 13 de noviembre de 2023. [En línea]. Disponible en: <https://mosquitto.org/>
- [16] J. Torres y A. López, "Implementación y análisis de sistemas de control y telemetría en invernaderos utilizando MQTT y el broker Mosquitto", *Revista de Investigación en Ingeniería Eléctrica, Electrónica y de Sistemas*, vol. 5, núm. 2, pp. 45–56, 2018.
- [17] R. Pérez y M. Gómez, "Análisis de rendimiento del broker Mosquitto en aplicaciones IoT", *Revista de Investigación en Tecnología de la Información y Comunicación*, vol. 9, núm. 2, pp. 78–89, 2020.
- [18] A. López y M. García, "Sistemas de telemetría: Principios y aplicaciones", en *Tecnologías de la Información y Comunicaciones en Ingeniería*, 2020, pp. 45–62.
- [19] R. Singh y S. Gupta, "Review of Message Queuing Telemetry Transport Protocol for Internet of Things", *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 4, núm. 2, pp. 1009–1013.