




Mosquitto: server and client implementation of the MQTT protocol

DOI: [10.21105/joss.00265](https://doi.org/10.21105/joss.00265)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Licence

Authors of JOSS papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Roger A Light

Summary

Mosquitto provides standards compliant server and client implementations of the MQTT messaging protocol. MQTT uses a publish/subscribe model, has low network overhead and can be implemented on low power devices such microcontrollers that might be used in remote Internet of Things sensors. As such, Mosquitto is intended for use in all situations where there is a need for lightweight messaging, particularly on constrained devices with limited resources.

The Mosquitto project is a member of the Eclipse Foundation

There are three parts to the project.

- The main `mosquitto` server
- The `mosquitto_pub` and `mosquitto_sub` client utilities that are one method of communicating with an MQTT server
- An MQTT client library written in C, with a C++ wrapper

Mosquitto allows research directly related to the MQTT protocol itself, such as comparing the performance of MQTT and the Constrained Application Protocol (CoAP) (Thangavel et al. 2014) or investigating the use of OAuth in MQTT (Fremantle et al. 2014). Mosquitto supports other research activities as a useful block for building larger systems and has been used to evaluate MQTT for use in Smart City Services (Antonić et al. 2015), and in the development of an environmental monitoring system (Bellavista, Giannelli, and Zamagna 2017). Mosquitto has also been used to support research less directly as part of a scheme for remote control of an experiment (Schulz, Chen, and Payne 2014).

Outside of academia, Mosquitto is used in other open source projects such as the openHAB home automation project and OwnTracks, the personal location tracking project, and has been integrated into commercial products.

References

- Antonić, A., M. Marjanović, P. Skočir, and I. P. Žarko. 2015. “Comparison of the CUPUS Middleware and MQTT Protocol for Smart City Services.” In *2015 13th International Conference on Telecommunications (ConTEL)*, 1–8. doi:10.1109/ConTEL.2015.7231225.
- Bellavista, Paolo, Carlo Giannelli, and Riccardo Zamagna. 2017. “The PeRvasive Environment Sensing and Sharing Solution.” *Sustainability* 9 (4): 585. doi:10.3390/su9040585.
- Fremantle, P., B. Aziz, J. Kopecký, and P. Scott. 2014. “Federated Identity and Access Management for the Internet of Things.” In *2014 International Workshop on Secure Internet of Things*, 10–17. doi:10.1109/SIoT.2014.8.
- Schulz, M., F. Chen, and L. Payne. 2014. “Real-Time Animation of Equipment in a Remote Laboratory.” In *2014 11th International Conference on Remote Engineering and*

Virtual Instrumentation (REV), 172–76. doi:10.1109/REV.2014.6784247.

Thangavel, D., X. Ma, A. Valera, H. X. Tan, and C. K. Y. Tan. 2014. “Performance Evaluation of MQTT and CoAP via a Common Middleware.” In *2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, 1–6. doi:10.1109/ISSNIP.2014.6827678.