Journal of Computer Science

Original Research Paper

# IOT Based Green House Monitoring System

**Tinu Anand Singh and J. Chandra**

*Department Of Computer Science, Christ University, Bangalore, India*

*Article history*
Received: 3-08-2017
Revised: 28-08-2017
Accepted: 01-02-2018

Corresponding Author:
J. Chandra
Department Of Computer
Science, Christ University,
Bangalore, India
Email: chandra.j@christuniversity.in

**Abstract:** With industrialization and continuously evolving climatic conditions, the urge to practice agriculture with the fusion of technology has become a necessity. In the era of Internet of Things where all eyes are witnessing the evolution of machine to machine interaction, there is also a lack of clarity in considering the type of protocol to be used in building a particular system like Green House. A green house is a regulated environment for agriculture where critical parameters like temperature, light, humidity, ph level of soil can be monitored with the help of sensor systems using Internet of Things protocols. Message Queue Telemetry Transfer protocol was chosen over Constrained Application Protocol and Extensible Messaging and Presence Protocol in the experiment conducted in terms of its light weight transmission, resource consumption and effectively providing the different quality of services to detect the temperature and humidity as well as the gas leaks encountered in a greenhouse environment.

**Keywords:** Greenhouse, Internet of Things, Message Queue Telemetry Transfer

## Introduction

As an immediate effect of worldwide environmental change, oil value climbs furthermore, budgetary hypothesis, food shortage has got to be one of today's most basic issues that people need to confront truly. Since the conventional farming is profoundly climate dependent, there lies an immediate urge to create and sustain an environment for round the year production using the scientific technology. Ahmed *et al.* (2016) describes continuous advancements in wireless domain and the development in the Internet of Things (IoT) have empowered the physical world to imperceptibly interlace with sensors, actuatorsand other computational components while continuously ensuring network connectivity. The persistently associated physical world with computational components shapes a savvy situation for a smarter environment. Thus providing an environment that aims and improve the capacities of its inhabitants in executing their undertakings.

Vogler *et al.* (2016) explains that since IoT contains many devices and correspondence concentrated design and so as to detect and control their surroundings, applications in the Internet of Things are required to coordinate and deal with a substantial number of heterogeneous gadgets, which generally range as straightforward sensors and actuators. As of late, however, devices that essentially sense the environment and activate elements are also obliged to execute in situations with constrained processing capacity, limited memoryand little bandwidth abilities. Vishwanath *et al.* (2016) explains that there is a need to standardize protocols conventions to empower all gadgets to speak with each other with different components along with various essential features like interoperability, latency, security, computation ability, quality of service and packet loss. In order to have a greenhouse monitoring system, the study of three important IOT protocols COAP, XMPP and MQTT is performed and it is observed that MQTT is significantly easier to implement by using the open source mosquitto broker on the raspberry pi.

The remaining paper is organized as follows: Firstly, the literature review in green house application is performed followed by a short survey. Then the focus is then given on the experimental environment and results are analyzed. Finally, conclusions and future enhancement is identified.

## Literature Review

Luzuriaga *et al.* (2015) presents an introduction to the protocol Message Query Telemetry Transport (MQTT).

Science Publications

The various levels of Quality of Service (QOS) is discussed and categorized into 3 types – (QoS = 0), (Qos = 1) and (Qos = 2) and described as message sent only once, at least once and exactly once respectively. Also a solution was proposed to make that guaranteed no information loss using the intermediate buffering technique. However the drawback was memory leaks when the buffer capacity was full and overloaded.

Stankovic (2014) does an analysis on the various research oriented concerns with the evolution of the Internet of Things (IOT). For the full fledge running of future IOT devices there is a need that arises to equip the networks and also ensure concerns like massive scaling using techniques like IPv6 and Low Powered Wide Area Network (LPWAN). The reliability of the devices and sensor devices will be based on its sensing and parameters like sleep/wakeup schedules using clock synchronization.

Singh *et al*. (2015) tried various mechanisms to make the MQTT network secure. An attribute based encryption is proposed by the author. Here the device at senders end encrypts data based on set of conditions based on terms of access policy. Similarly at the receiver end the device is able to decrypt the ciphertext, if it satisfies access policy that are expressed as a predicate with set of attributes and boolean constructs (ORand, NOT). The feasibility to enable secure communication in IOT devices using Publish-Subscribe architecture.

Hunkeler *et al.* (2008) tested the MQTT-S over wireless networks. Client side or publisher was written in C using the Zigbee network transmission. However, it was found that the Client API automatically sends keep-alive messages even though a device has failed and no longer sends messages; the Client API continued to ping requests. Also in the case of node failure it was not possible to trigger the "will message" and with QoS 1 or 2 it becomes impossible to send the message from the broker to the clients.

Kim *et al.* (2015) understood the need for unique-IoT administration at home. Also, the basic and powerful approaches to oversee different machines and gadgets. Thus, the home environment needs a door that gives dynamical gadget enrollment and revelation using the light-weight MQTT protocol. The proposed system allowed to develop a energy saving system.

Hemraj and Sukesha (2014) proposed the Adaptive On Demand Transmission Power Control protocol in the greenhouse application on the capsicum crop with a threshold temperature range set to 18-25°C. It was observed to increase the battery efficiency in wireless networks but the reading were hardware dependent.

Samal and Pati (2014) performed the data acquisition and data logging of soil ph level and temperature in a greenhouse environment using the LabVIEW interface and also proposed the future enhancement of TCP/IP and wireless data acquisition.

## Methodology

The most critical parameter for a greenhouse is humidity. The greenhouses humidity is thus measured in terms of the relative humidity which is a mixture of ratios of water vapor ($H_2O$) $P_w$ in the mixture to the saturated vapor pressure of water $P_s$ at a given temperature. In general, relative humidity is expressed in percentage of the actual vapor density by the saturation vapor density. The temperature inside the greenhouse is yet critical as it directly affects the photosynthetic and transpiration processes of the crops. Similarly, $MQ_5$ and $MQ_7$ gas sensors absorb the gases in the surrounding greenhouse environment and indicate the peak rise in readings when a gas leak or incident is recognized by alerting the user with an alarm signal to take the necessary action. The transmission of information takes place from the sensor publishing the readings to the Mosquitto broker and the subscribers of the greenhouse topic are instantaneously alerted.

The earliest protocol Extensible Messaging and Presence Protocol (XMPP) designed by the open source community enabled the near-real-time communication. However to function under the constrained nodes and constrained networks, the Constrained Application Protocol (CoAP) request/response system model was proposed for IOT. Similarly Message Queue Telemetry Transport (MQTT) was released by IBM and designed in regards with M2M communications to provide flexibility and a publish/subscribe model unlike the request/ response method.

### *COAP*

Thamer *et al.* (2013) proposes Constrained Application Protocol (CoAP) which is a request/response protocol that runs over UDP to fulfil the purpose of lightweight implementation in IOT by removing the TCP/IP overhead. Keoh *et al.* (2014) explains that COAP uses the HTTP commands GET, POST, PUT and DELETE in a client-server architecture. Since it runs on UDP which is unreliable method of communication and security becomes a concern.

CoAP does not include any built-in security features. Thamer explains that to secure CoAP transactions in the IOT scenario the Datagram Transport Layer Security (DTLS) is embedded on top of it that further provides administration in terms verification, data integrityand cryptographic calculations. The core advantage of CoAP is that it supports the data distribution technique of multicast where a message can be distributed across various devices at the same time. However, Raza *et al*. (2013) confirm that DTLS doesn't support multicast and when embedded on COAP and it increases the computation and network congestion with the drastic increase in packet size that are to be transmitted.

### *XMPP*

The benefit of XMPP as the default convention for IoT is that it is a standardized and established convention intended for real-time information transmission that can be utilized without the requirement for a middleware or convention portals. XMPP offers a rich assortment of open source programming for servers, customersand libraries supporting a few working frameworks, going from desktop PCs to portable substances, hence effortlessly interfacing different gadgets and diminishing developing and testing costs. Bendel *et al*. (2013) explain Extensible Messaging and Presence Protocol (XMPP) was designed to work with synchronous request/response and has undergone many updates to provide the synchronous publish/subscribe method as well. It also provides the built-in DTLS security to support the secure Machine to Machine (M2M) communication enhancing the reliability of transmission. However, used power consumption significantly increases with the overhead of XML(eXtensible Markup Language) parsing that needs additional computational ability. To ensure security XMPP has built in TLS/SSL security but lacks the option of quality of services as desired by the system environment and impractical for the machine to machine communication.

### *MQTT*

Hence, Message Queue Telemetry Transport (MQTT) was released by IBM and designed in regards with M2M communications to provide flexibility. Lee *et al*. (2013) discuss that MQTT is bidirectional, asynchronous, publish/subscribe protocol unlike the request/response and hence decreasing the bandwidth usage and computational resources making it ideal for the IOT scenario.

MQTT permits gadgets to publish data to specific topic onto a broker. Then the broker delivers the data out to those clients that have subscribed on to the client's topic. Clients can subscribe to a particular level of a topic pecking order a special case character to subscribe to different levels of hierarchy. MQTT is used for making decision for remote systems that faces fluctuating levels in terms of idleness because of intermittent transmission requirements or inconsistent associations. Ought to the association from a subscribing client onto the intermediary gets broken, then the broker will support messages and deliver them to the supporter when it is back on the web. Ought to the association from the distributing client to the broker be separated without notice, the specialist can close the association and deliver endorsers a reserved message with guidelines from the distributer.

A MQTT session is separated into four phases: association, validation, correspondence and end. A customer begins by making a TCP/IP association with the representative by either utilizing a standard port or a custom port characterized by the merchant's administrators. While interfacing, perceive that the server may proceed with an old session if furnished with a re-utilized customer personality.

MQTT is known as a lightweight convention since all messages have a little code impression. Every message comprises of a settled header (2 bytes), a discretionary variable header, a message payload that is restricted to 256 MB of data and a Quality of Service (QoS) level. The three distinctive Quality of Service levels decide how the substance is overseen by the MQTT convention.

### *Proposed System*

The proposed system closely monitors and controls the microclimatic parameters of a greenhouse for the cultivation of crops or specific plant species which could maximize their production over the whole crop growth season and to eliminate the difficulties involved in the system by diminishing human mediation to the best conceivable degree.

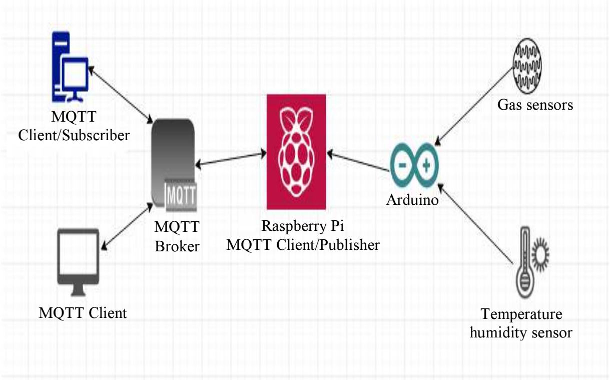

**Fig. 1:** Proposed Architecture for IOT Based GHMS

The Fig. 1 depicts the architecture of the proposed IOT based GHMS where the gas sensors demonstrate the $MQ_5$ and $MQ_7$ gas sensors along with the temperature and humidity sensor that is measured using the $DHT_{11}$ sensor. These sensors that sense the environment are connected to the Arduino microcontroller pass the collected information to the Raspberry Pi. On the Raspberry Pi, The current research work has MQTT Broker Mosquitto installed that establishes the connection to various client and subscribers. The proposed system closely monitors and controls the microclimatic parameters of a greenhouse for the cultivation of crops or specific plant species which could maximize their production over the whole crop growth season and to eliminate the difficulties involved in the system by diminishing human mediation to the best conceivable degree.

In the experiment conducted an open source Mosquitto broker is installed on Raspberry Pi that performs the publish and subscribe method for the MQTT protocol. DHT11 sensor is used to sense the temperature and humidity of the environment.

The most critical parameter for a greenhouse is humidity. The greenhouses humidity is thus measured in terms of the relative humidity which is a mixture of ratios of water vapor ($H_2O$) Pw in the mixture to the saturated vapor pressure of water Ps at a given temperature. Relative humidity is usually expressed in percentage of the actual vapor density by the saturation vapor density. The temperature inside the greenhouse is yet critical as it directly affects the photosynthetic and transpiration processes of the crops.

Similarly, MQ5 and MQ7 gas sensors absorb the gases in the surrounding greenhouse environment and indicate the peak rise in readings when a gas leak or incident is recognized by alerting the user with an alarm signal to take the necessary action. The transmission of information takes place from the sensor publishing the readings to the Mosquitto broker and the subscribers of the greenhouse topic are instantaneously alerted.

## Results and Discussion

In the experiment conducted with the MQTT protocol on the raspberry Pi with the various sensor systems we attained the following results:

**Table 1:** DHT11 Readings

| Humidity (%) | Temperature (Celsius) |
|---|---|
| 32 | 29 |
| 33 | 28 |
| 33 | 27 |
| 34 | 26 |
| 34 | 25 |
| 36 | 20 |
| 36 | 19 |
| 38 | 18 |
| 38 | 18 |
| 39 | 15 |

The Table 1 represents the values of DHT11 sensor readings are recorded for over 80 trials and their average is taken into consideration to plot a graph as depicted in the below figure Humidity and Temperature.

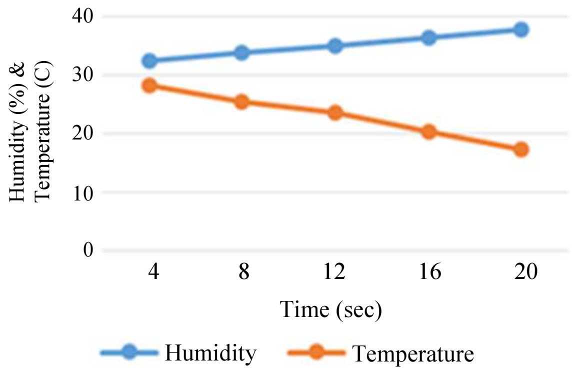

**Fig. 2:** Humidity and temperature of DHT11

The Fig. 2 depicts the sensor readings are measured in seconds along the Y-axis and the humidity and temperature is represented in percentage and degree Celsius along the X-axis respectively. It was observed that there was little or no packet loss or delay in delivering the reading of the sensor to the MQTT broker and delivering the same instantaneously to the subscriber clients.

**Table 2:** MQ5 and MQ7 sensor readings

| Gas Leaks | Carbon Monoxide |
|---|---|
| 81 | 111 |
| 83 | 113 |
| 89 | 113 |
| 93 | 115 |
| 96 | 118 |
| 189 | 181 |
| 531 | 223 |
| 515 | 207 |
| 268 | 320 |
| 222 | 316 |

The Table 2 represents the values of MQ5 and MQ7 sensor readings are recorded for over 90 trials and their average is taken into consideration to plot a graph as depicted below. Through this research work, it is observed that these results based on creating an artificial scenario where there is a sudden gas leakage caused by burning of woods.

These reading of the environment recorded by MQ5 is with respect to the gas leakage that includes H2, CH4, LPG. Similarly the carbon monoxide levels that are measured by MQ7 are also seen increasing and saturating after a particular time interval.

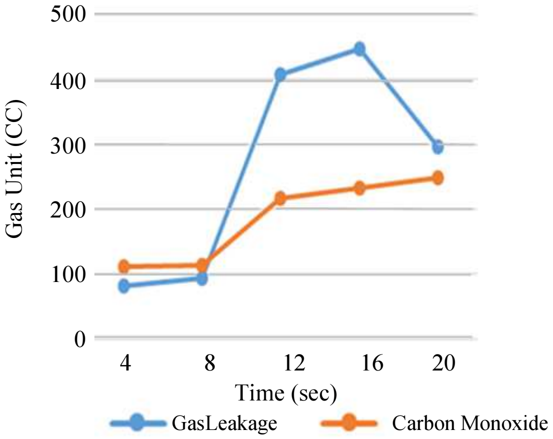

**Fig. 3:** Gas reading for MQ5 and MQ7

The Fig. 3 represent the peak rise in gas readings as recorded by the MQ5 and MQ7 gas sensor and these are immediately transferred from the Arduino board to the MQTT broker and instantly displayed on the subscribers without any delay. Hence alerting the user of the immediate action required to control the gas leakage.

## Conclusion

In this research work focus is on providing an effective solution to adapt MQTT in greenhouse application scenarios. The advantage is that developers do not have to explicitly consider the changes in the point of attachment to the network. The current work is based on low power consumption using MQTT protocol and the low maintenance making it compact, portable and robust.

Also, using appropriate QoS level for different payloads, network environment can further be controlled to provide optimum results for the crops. The future enhancement includes multiple raspberry pi connected to monitor the data on a larger scale and its implementation to large agricultural areas with other sensors like ph sensor etc.and to experiment under various QoS levels and payloads and to choose the most efficient QoS level of MQTT protocol setting in the Greenhouse environment. Detailed analysis of MQTT Broker server performance, communication among multiple clients and throughput rates considering different numbers of topics needs to be studied when implementing on large scale environment.

## Acknowledgement

The authors would like to thank Prof. Joy Paulose, HOD, Department of Computer Science, Christ University Bangalore, India for constantly motivating us in the entire process of our research.

## Author’s Contributions

All authors equally contributed in this work.

## Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other author have read and approved the manuscript and no ethical issues involved.

## References

Ahmed, E., I. Yaqoob, A. Gani, M. Imran and M. Guizani, 2016. Internet-of-things-based smart environments: State of the art, taxonomyand open research challenges. IEEE Wireless Communi., 23: 10-16. DOI: 10.1109/MWC.2016.7721736

Bendel, S., T. Pringer, D. Schuster, A. Schill and R. Ackermann *et al*., 2013. A Service Infrastructure for the Internet of Things based on XMPP. Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops, March 18-22, IEEE Xplore press, San Diego, pp: 385-388. DOI: 10.1109/PerComW.2013.6529522

Hemraj and Sukesha, 2014. Power estimation and automation of green house using wireless sensor network. Proceedings of the 5th International Conference-Confluence the Next Generation Information Technology Summit, Sept. 25-26, IEEE Xplore press, Noida, India. DOI: 10.1109/CONFLUENCE.2014.6949374

Hunkeler, U., H.L. Truong and A. Stanford-Clark, 2008. MQTT-S – a publish/subscribe protocol for wireless sensor networks. Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops, Jan. 6-10, IEEE Xplore press, Bangalore. DOI: 10.1109/COMSWA.2008.4554519

Keoh, S.L., S.S. Kumar and H. Tschofenig, 2014. Securing the internet of things: A standardization perspective. IEEE Internet Things J., 1: 265-275. DOI: 10.1109/JIOT.2014.2323395

Kim, S.M., H.S. Choi and W.S. Rhee, 2015. IoT Home Gateway for Auto-Configuration and Management of MQTT devices. Proceedings of the IEEE Conference on Wireless Sensors, Aug. 24-26, IEEE Xplore press, Melaka, Malaysia. DOI: 10.1109/ICWISE.2015.7380346

Lee, S., H. Kim, D.K. Hong and H. Ju, 2013. Correlation analysis of MQTT loss and delay according to QoS level. Proceedings of the International Conference on Information Networking, Jan. 28-30, IEEE Xplore press, Bangkok, Thailand, pp: 714-717. DOI: 10.1109/ICOIN.2013.6496715

Luzuriaga, J.E., J.C. Cano, C. Calafate, P. Manzoni and M. Perez *et al*., 2015. Handling mobility in IoT applications using the MQTT protocol. Proceedings of the Internet Technologies and Applications, Sept. 8-11, IEEE Xplore press, Wrexham, UK. DOI: 10.1109/ITechA.2015.7317403

Raza, S., H. Shafagh, K. Hewage, R. Hummen and T. Voigt, 2013. Lithe: Lightweight Secure CoAP for the Internet of Things. IEEE Sensors J., 13: 3711-3720. DOI: 10.1109/JSEN.2013.2277656

Samal, N. and U.C. Pati, 2014. Multi-channel data acquisition and data logging for green house application. Proceedings of the IEEE Students' Conference on Electrical, Electronics and Computer Science, March 1-2, IEEE Xplore press, Bhopal, India. DOI: 10.1109/SCEECS.2014.6804507

Singh, M., M.A. Rajan, V.L. Shivraj and P. Balamuralidhar, 2015. Secure MQTT for internet of things (IoT). Proceedings of the Fifth International Conference on Communication Systems and Network Technologies, April 4-6, IEEE Xplore press, Gwalior, India. DOI: 10.1109/CSNT.2015.16

Stankovic, J.A., 2014. Research directions for the internet of things. IEEE Internet Things J., 1: 3-9. DOI: 10.1109/JIOT.2014.2312291

Vishwanath, S.K., C. Yen, W. Tushar, W.T. Li and C.K. Wen *et al*., 2016. System design of the internet of things for residential smart grid. Proceedings of the IEEE Wireless Communi., 23: 90-98. DOI: 10.1109/MWC.2016.7721747

Vogler, M., J. Schleicher, C. Inzinger and S. Dustdar, 2016. Optimizing Elastic IoT Application Deployments. IEEE Trans. Services Comput. DOI: 10.1109/TSC.2016.2617327