IST-Africa
IST-Africa 2018 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2018
ISBN: 978-1-905824-60-1

An Analysis of CoAP as Transport in an Internet of Things Environment

Louis COETZEE[1], Dawid OOSTHUIZEN[2], Buhle MKHIZE[1]
[1]*CSIR Meraka Institute, Pretoria, South Africa*
Email: louis.coetzee@csir.co.za, bmkhize@csir.co.za
[2]*Nearmap Australia Pty Ltd, Sydney, Australia*
Email: dawid.oosthuizen@nearmap.com

Abstract: Internet of Things (IoT) is a concept where the physical world is integrated into the digital world. In IoT, Internet connected devices are able to observe and act. Furthermore, data observations from these devices are linked with cloud based digital services to enable smart decision-making. IoT is seen as the underpinning for smart cities, with multiple services including intelligent traffic management, and effective resource management (e.g. energy or water). The benefits and potential of IoT are clear, resulting in significant uptake. However, numerous factors still hamper its growth. These factors include performance of networks and edge sensing devices, up to developing appropriate applications, all done in a secure and reliable manner. Data communication can typically occur over various networks and protocols, depending on the context of deployment and operation. This paper focuses on validating the Constrained Application Protocol (CoAP) in a low power personal area network to determine if CoAP is an effective application protocol in an IoT environment. Experiments were conducted in the context of the TRESCIMO testbed through which CoAP enabled devices and gateways communicate to IoT platforms. The experiments focused on measuring the efficiency of communication in an environment with poor connectivity. Experimental results show that CoAP is an efficient transport in low signal strength environments.

Keywords: Internet of Things, Constrained Application Protocol, CoAP, Smart City, Smart decision-making, TRESCIMO.

1. Introduction

Internet connected devices (with the ability to sense and also actuate) are becoming ubiquitous. Linking these devices with smart, cloud based decision-making services allow for sense-making that can create impact (either financially and/or socially). This fusion of technologies is commonly referred to as the Internet of Things (IoT) [1]. It is projected that the number of Internet devices will number into the billions and also that business stemming from IoT will be billions of dollars [2].

The benefits of IoT are clear. With contextual information acquired from the physical environment, smarter decisions spanning multiple domains can be made through a variety of applications [3]. However, the uptake of IoT has been slow and mostly limited to silos. A number of factors have impacted on the slow uptake. These include the challenge in providing a secure solution, the difficulty in choosing the "right" middleware (e.g. the market is flooded with a variety of middleware platforms, each providing its own approach [4]), and ensuring appropriate end-to-end network connectivity [5].

In a typical IoT architecture data observations are sent via a gateway over the Internet ("north" interface) to an implementation of an IoT middleware technology. From the middleware data is routed to applications. This process is depicted in Figure 1. Observations are often collected from a personal area network on the "south" interface of

the gateway. These personal area networks can have low power requirements. The associated sensing device has to either harvest energy or is dependent on batteries. This in turn impacts on the amount of data that can be communicated within a timespan, and the distance the sensing device can be from the gateway as the energy consumption of the communication stack has to be limited, which in turn impacts on the transmission signal strength.

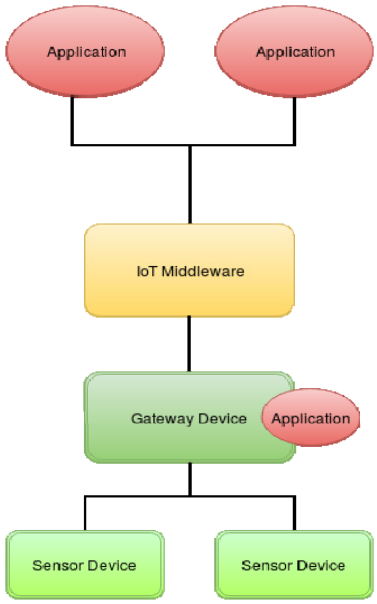

Figure 1 Generic Internet of Things Architecture

A number of different protocols have been developed for application in low power environments. These include MQTT [6], MQTT-SN [7] and Constrained Application Protocol (CoAP) [8],[9]. MQTT and MQTT-SN implement a publish/subscribe architecture, while CoAP follows a client/server approach. Table 1 presents an analysis of the key characteristics of these three popular IoT protocols [10]. Both MQTT-SN and CoAP are well suited to constrained devices. MQTT is often better suited to more powerful devices (as it requires a full network stack). MQTT uses TCP which is considered to be slightly heavier in bandwidth utilisation due to the TCP "handshaking" to ensure reliable connectivity. In addition, it has to keep an open socket to the broker, which can be problematic in an environment with poor connectivity, or where power is a key consideration. MQTT-SN and CoAP use UDP (allowing for significantly smaller header sizes), with message delivery ensured at application level. MQTT-SN based devices register their topics at the broker, thus avoiding sending a long topic as is the case with MQTT, instead sending only a topic id. MQTT allows for many-to-many message delivery, while CoAP is mostly client/server with point delivery of a message. CoAP provides for dynamic resource discovery by publishing its "well-known/core" which allows other devices to discover and interact with a specific resource. Topics on the broker can be discovered and filtered in MQTT and MQTT-SN to obtain access to a specific observation stream. CoAP supports a form of "broadcast" to multiple clients through the UDP "multicast" capability. Furthermore CoAP provides an "observe" function, which allows clients to register to a resource on a device, and subsequently receive observations from the device without having to interrogate the device again.

The three mentioned protocols all serve IoT and IoT based networks. MQTT-SN and CoAP are positioned for low-power and constrained environments, while MQTT is better suited for better resourced environments. Choosing a specific technology is very much dependent on the environment and associated constraints. Viewing the above, CoAP is a good choice in a low power environment, while MQTT is well suited for environments

requiring reliable data transfer. Both MQTT and MQTT-SN are good choices for observation delivery, while CoAP provides excellent support to interact in a bi-directional manner with a device via the REST approach. The authors regard this function as a unique strength when seamless control (actuation) on a device is required (the devices used in this paper require seamless actuation). In the context of this paper we will analyse CoAP.

Table 1 MQTT, MQTT-SN, CoAP Characteristic comparisons

	MQTT	**MQTT-SN**	**CoAP**
Architecture	Publish/Subscribe via broker (middleware)	Publish/Subscribe via broker (middleware)	Client-Server (URI-based)
Quality of Service	QoS 0: Fire-and-forget; QoS 1: Message delivered at least once; QoS 2: Message delivered exactly once	QoS 1: Message confirmed by receiver with "Ack"; QoS 2: Message delivered exactly once	Confirmable (message confirmed by receiver with "Ack"); Non-confirmable (fire-and -forget)
Security	Transport Layer Security (TLS)	Depends on network technology	Datagram Transport Layer Security (DTLS)
Transport	TCP	UDP	UDP

The Constrained Application Protocol (CoAP) has only recently been standardised (2014). As a consequence not many real-world deployments have been done where the suitability and efficiency of CoAP could be validated. This paper analyses the appropriateness of CoAP in terms of data communication in a constrained and poor connectivity environment. It aims to determine if CoAP is a suitable protocol for IoT deployments. In order to validate the suitability of CoAP for IoT environments, an IoT testbed and its related components developed under the EU funded project "TRESCIMO", were used [11], [12]. A CoAP based energy sensing device (plug) was configured to communicate to a CoAP enabled gateway. Data observations were communicated from the sensing device to the gateway. The effectiveness of communicating the observations at different received signal strengths was determined to answer the question if CoAP can successfully be used in IoT deployments.

The paper is structured as follows: Section 2 describes the components within a CoAP based personal area network. Section 3 presents the experimental configuration, while Section 4 presents results in evaluating data communication and the suitability of CoAP for IoT deployments. Section 5 presents our conclusions.

2. Constrained Application Protocol

The Constrained Application Protocol (CoAP) was published as a full IETF Internet standard in 2014. It follows a client/server approach, where the CoAP server publishes a set of "resources" (IP addressable sensors and actuators). Clients follow a "Representational State Transfer" (REST) approach where data is obtained from a resource using a "GET" request, and where actuation and control commands can be sent via a "PUT" or "POST". Clients can also "subscribe" to a resource and thus receive continuous updates if a resource is updated [8], [9].

Figure 2 presents a view of the network decomposed into a number of layers. The IEEE Standard 802.15.4 provides for Low-Rate Wireless Networks for the physical network layer [13]. 6LowPAN provides for a compression of IPV6 packets, in order for packets to be sent and received over a IEEE 802.15.4 network [14]. The IPV6 layer (a new version of the Internet Protocol) standardises the delivery of packets from a host to destination based on an IP address. The User Datagram Protocol (UDP) is a core component of packet delivery as associated with IPV6 [15]. It does not provide for guaranteed delivery (in contrast to

Transmission Control Protocol (TCP) which does). The final layer in the network stack is CoAP [8]. CoAP provides for a low overhead application protocol, specifically developed for machine-to-machine and associated IoT networks.

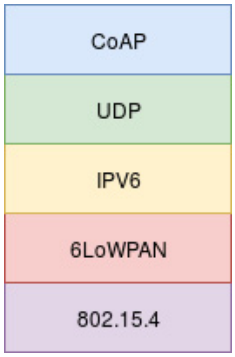

Figure 2 Network layers

As CoAP is UDP based, it is believed that the networking overhead typically associated with TCP would be negated. To facilitate message delivery confirmations, a UDP based "confirmation" and "retry" model is included.

3. Experimental Environment

Experiments focusing on the efficiency of communication in an environment with poor connectivity were conducted in the context of the TRESCIMO testbed. TRESCIMO utilises a "platform-as-a-service" methodology with a variety of components (i.e. a Smart City Platform, a oneM2M compliant communication platform and CoAP enabled sensing and actuation devices) to create the testbed [11], [12], [16]. The components making up the testbed are open and configurable, which allows for the introduction of new experiments to answer a broad set of research questions. The TRESCIMO testbed has specifically been created to experiment and validate smart solutions with potential for impact e.g. smart energy management [11], [12]. Figure 3 presents the TRESCIMO reference architecture. The experiments conducted in this paper focuses on the connectivity between the "Smart City Platform Edge" and the "Active Device" components.

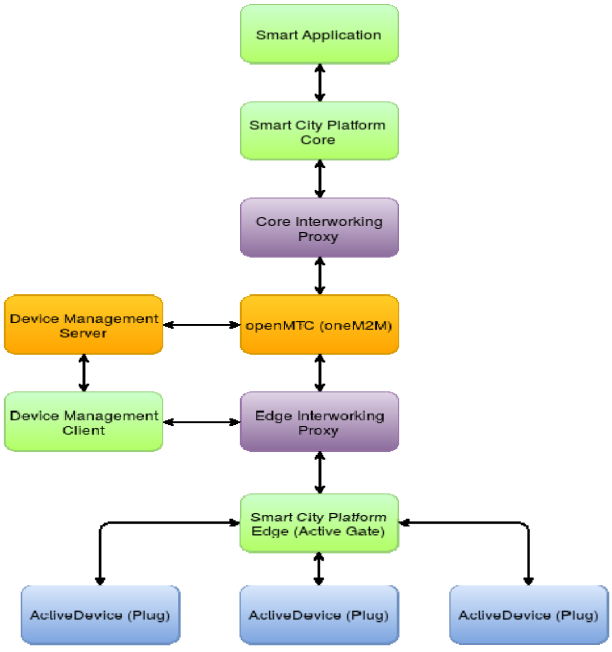

Figure 3 TRESCIMO Reference Architecture

The experiments were conducted at the hand of data observations from devices and services from a true IoT installation (in the form of a Smart Kitchen where a number of appliances have been instrumented to measure energy consumption and also provide for remote actuation and control). The Smart Kitchen is one of the use-cases in the TRESCIMO testbed.

The testbed can be scaled in terms of the number of connected devices, as well as in a spatial manner, linking components hosted on different continents. This freedom in choice enables Future Internet based experimentation [17]. A key component in the testbed is that of a personal low power network, linking devices with low power requirements to the gateway. For the testbed this was done in the form of purpose built CoAP "Active" devices (i.e. ActivePlug and ActiveGate) [18].

Figure 4 presents the gateway device, named as the ActiveGate, with Figure 5 depicting the sensing device, referred to as the ActivePlug. The two devices (ActivePlug and ActiveGate) communicate using a 2.4GHz 802.15.4 radio module based on the STM32W108 System-on-Chip (SoC) from STMicroelectronics.

ActiveGate is a processing and routing platform that consists of the following:

- An Odroid-U3+ single board computer, with a 1.7GHz Exynos4412 Prime ARM Cortex-A9 quad-core processor, 2GB RAM, and various external interfaces.
- A power supply and I/O board.

Figure 4 Constrained Application Protocol Gateway (ActiveGate)

The ActivePlug is an energy management device, with the following features:

- STPM01 metrology circuitry for measuring voltage, current, power, line frequency as well as active, reactive, apparent, and fundamental energy consumption.
- ARM Cortex-M4 microcontroller for managing the metrology, load switching, and interface functions.
- Indicators for current power load, switching status, and device status.
- USB interface for diagnostics and programming, electrically isolated from the rest of the board.
- Three general purpose analogue inputs, electrically isolated from the rest of the board, rated for 0 to 5V DC.

Figure 5 Constrained Application Protocol Sensing Device (ActivePlug)

4. Results

The primary question related to using CoAP in a low power network environment is its reliability measured against the signal strength. Can the data observations still be effectively communicated when the connectivity is poor? Using the TRESCIMO experimental environment, measurements were obtained where a set number of observations were sent between the ActivePlug and the ActiveGate. By moving the ActivePlug further away from the ActiveGate, the signal strength would drop. As indicator of the effectiveness and appropriateness of CoAP, the time taken to deliver the set number of observations was used. For the experimental setup, 5 observations from the sensing device were expected within 25 seconds at the gateway. If the time taken to deliver the total number of messages increased significantly from the 25 seconds, it can be inferred that the CoAP message delivery has been negatively impacted.

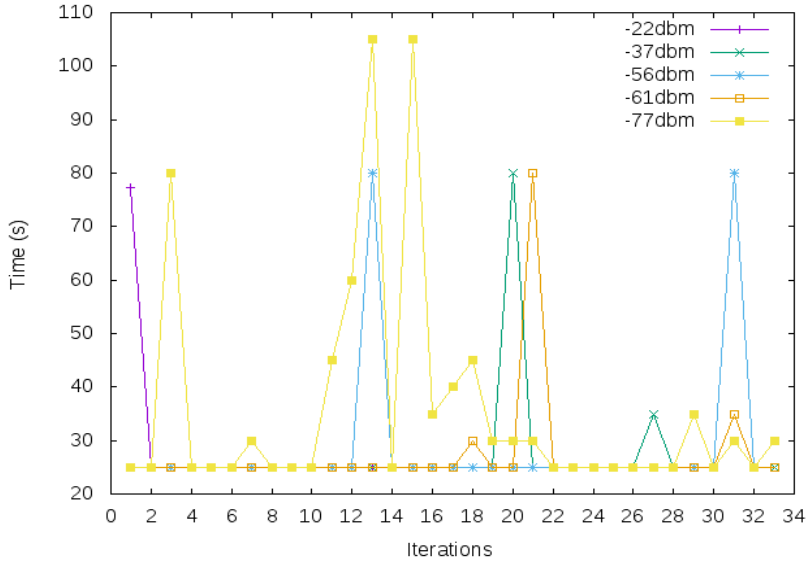

Figure 6 Observation delivery time at different signal strengths

The measured delivery times against the number of iterations (where each iteration comprises of sending the set number of data observations) are depicted in Figure 6. The measurements were obtained at different signal strengths (–22dbm to -77dbm). At -22dbm almost all observations were received by the gateway within the desired timeframe. At other signal strength levels, observations were mostly received within the allocated time. With the continued decrease in signal strength, connectivity was lost more frequently, or packet loss increased significantly (requiring more "retries" from CoAP). It should be noted that even at a very poor Received Signal Strength Indicator (RSSI), observations were still delivered from the sensing device to the gateway device. This is a testament to the CoAP protocol, indicating that it can still perform at an adequate level, even within a poor and noisy radio frequency environment.

5. Conclusions

Experiments were conducted to verify if CoAP is an effective protocol for wireless machine-to-machine and IoT networks. The experiments were conducted on an IoT testbed (TRESCIMO) that supports a "plug-and-play" experimental environment. In the context of this paper, energy sensing devices (plugs) communicated to the gateway, over a constrained network, using CoAP as protocol. Signal strength between the devices was changed for the delivery of messages. The time taken to deliver the messages at a specific signal strength was compared to the baseline delivery time at a good quality signal strength.

The results indicated that CoAP is an efficient protocol and is able to operate in a low signal strength environment. At all signal strengths, CoAP based observations were delivered. However, at poorer signal strengths, it was noticeable that the delivery of observations was negatively impacted more often than at better signal to noise ratios. The fact that observations were still delivered, indicates that CoAP can successfully be used in different environments, including where connectivity is challenging.

References

[1] I. T. Union, *ITU Internet Reports 2005: The Internet of Things. Executive Summary*. Geneva: ITU, 2005 Available online. http://www.itu.int/osg/spu/publications/internetofthings/

[2] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin, and D. Aharon, "The Internet of Things: Mapping the Value Beyond the Hype," McKinsey Global Institute, 2015 Available online. http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/the-internet-of-things-the-value-of-digitizing-the-physical-world

[3] L. Coetzee and J. Eksteen, "The Internet of Things – Promise for the future? An Introduction," in *IST-Africa 2011 Conference Proceedings*, 2011.

[4] A. Gluhak and O. Vermesan, "H2020 – UNIFY-IoT Project, Deliverable D03.01, Report on IoT platform activities," 2016 Available online. http://www.unify-iot.eu/wp-content/uploads/2016/10/D03_01_WP02_H2020_UNIFY-IoT_Final.pdf

[5] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelffle, *Vision and Challenges for Realising the Internet of Things*. Brussels: European Commision- Information Society and Media DG, 2010 Available online. http://www.internet-of-things-research.eu/pdf/IoT_Clusterbook_March_2010.pdf

[6] A. Banks and R. Gupta, "OASIS Standard MQTT Version 3.1.1." Oct-2014 Available online. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

[7] A. Stanford-Clark and H. L. Truong, "MQTT For Sensor Networks (MQTT-SN) Protocol Specification." Nov-2013 Available online. http://mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf

[8] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP) RFC 7252." Jun-2014 Available online. http://www.rfc-editor.org/info/rfc7252

[9] Z. Shelby, "Constrained RESTful Environments (CoRE) Link Format (RFC 6690)." Aug-2012 Available online. http://www.rfc-editor.org/info/rfc6690

[10] M. H. Amaran, N. A. M. Noh, M. S. Rohmad, and H. Hashim, "A Comparison of Lightweight Communication Protocols in Robotic Applications," *Procedia Computer Science*, vol. 76, pp. 400–405, 2015 Available online. http://www.sciencedirect.com/science/article/pii/S1877050915038193

[11] "TRESCIMO – Testbeds for Reliable Smart City Machine to Machine Communication." Available online. http://trescimo.eu

[12] M. Barros, A. Gavras, A. A. Corici, R. Steinke, N. Mukudu, N. Ventura, J. Mwangama, D. Nehls, B. Riemer, L. Coetzee, D. Oosthuizen, M. Catalan, L. Herrera, J. Castells, R. Yatom, H. Madhoo, and T. Willemse, "TRESCIMO M2M-IoT Testbed," in *European Conference on Networks and Communications 2016: Posters (EuCNC2016-Posters)*, 2016, pp. 304–308.

[13] IEEE, "IEEE Std 802.15.4 – IEEE Standard for Low-RAte Wireless Networks." Available online. https://standards.ieee.org/findstds/standard/802.15.4-2015.html

[14] IETF, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN)." 2017 Available online. https://tools.ietf.org/html/rfc8066

[15] IETF, "User Datagram Protocol." Aug-1980 Available online. https://tools.ietf.org/html/rfc768

[16] L. Coetzee, M. Catalan, J. Paradells, A. Gavras, and M. J. Barros, "Building the Future Internet through FIRE 2016, FIRE Book: A Research and Experiment based Approach," M. Serrano, N. Isaris, H. Schaffers, J. Domingue, M. Boniface, and T. Korakis, Eds. River Publishers, 2016, pp. 211–241 Available online. https://drive.google.com/file/d/0Bw2qexyLJF8QUkJfTEJtV0F3ejQ/view

[17] A. Gavras, A. Karila, S. Fdida, and et. al., "Future internet research and experimentation: the fire initiative," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 3, pp. 89–92, 2007.

[18] A. Corici, "TRESCIMO – Deliverable D3.3: Integrated Prototype v2." Dec-2015 Available online. http://trescimo.eu