Performance evaluation on internet of things protocols to identify a suitable one for millions of tiny internet nodes

S.P. Raja*

Department of Computer Science and Engineering,
Vel Tech Rangarajan Dr. Sagunthala R&D Institute of
Science and Technology,
Avadi, Chennai, Tamil Nadu, India
Email: avemariaraja@gmail.com
*Corresponding author

T. Sampradeepraj

Department of Computer Science and Engineering,
Sethu Institute of Technology,
Kariapatti, Pulloor, Tamil Nadu, India
Email: sampradeepraj@gmail.com

N. Narayanan Prasanth

Department of Database Systems,
School of Computer Science and Engineering,
Vellore Institute of Technology,
Vellore, Tamil Nadu, India
Email: narayana.prasanth@gmail.com

Abstract: Recent days throughout the world, cities are facing challenges such as population, crimes, transportation, job opportunities, economic growth, environmental sustainability, waste management, traffic flow and energy consumption. In India, many cities are facing many serious issues in the form of air pollution. Considering the above challenges, many cities are spending on information and communication technology to make a sustainable environment which is necessary for the mankind. Internet of things (IoT) has been focused as a growing technology for solving the above social problems and to make a city as smart. Smart cities have been viewed to integrate multiple information and communication technology solutions to enhance efficiency and economical value. Many IoT protocols like low power wireless personal area networks (6LoWPAN), message queue telemetry transport (MQTT), constrained application protocol (CoAP), extensible messaging and presence protocol (XMPP), routing protocol for low power and lossy networks (RPL) and ZigBee protocol are used to make a smart environment. Billions of tiny devices are connected together to make a smart platform. In this paper, IoT protocols that can be suited for smart environments are implemented to find the suitable protocol for connecting millions of tiny internet nodes.

Keywords: internet of things protocols; 6LoWPAN; message queue telemetry transport; MQTT; constrained application protocol; CoAP; extensible messaging and presence protocol; XMPP; RPL; ZigBee.

Reference to this paper should be made as follows: Raja, S.P., Sampradeepraj, T. and Prasanth, N.N. (2019) 'Performance evaluation on internet of things protocols to identify a suitable one for millions of tiny internet nodes', *Int. J. Intelligent Internet of Things Computing*, Vol. 1, No. 1, pp.2–22.

Biographical notes: S.P. Raja completed his BTech in Information Technology in 2007 from the Dr. Sivanthi Aditanar College of Engineering, Tiruchendur, ME in Computer Science and Engineering in 2010 from the Manonmaniam Sundaranar University, Tirunelveli and PhD in 2016 from the Manonmaniam Sundaranar University, Tirunelveli. His area of interest is image processing and cryptography. He is having more than eleven years of teaching experience. He published seven papers in international journals, 25 in international conferences and 12 in national conferences.

T. Sampradeepraj received his BE in Computer Science and Engineering from the Bharathiyar University, Tamilnadu, India in 2002, ME in Wireless Technologies from the Thiagarajar College of Engineering, Anna University, Tamilnadu, India in 2005 and his PhD from the Manonmaniam Sundarnar University, Tirunelveli. His current research interests include wireless sensor networks, network coding and security.

N. Narayanan Prasanth is working as an Associate Professor at the Department of Database Systems, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamilnadu, India. He completed his PhD from the Manonmaniam Sundaranar University, Tirunelveli, India in 2017. His research area includes network switch scheduling, buffering issues, switching architecture, and distributed systems.

1 Introduction

Smart environment projects are envisioned and internet of things (IoTs) gives the initiatives all over the world. The IoTs for smart cities requires accessible public data and systems which makes companies to develop new services and functionalities. The vision of Smart cities is to improve the quality of urban services by make use of information and communication technologies. Smart cities make us more safe and digitised. IoT in smart cities makes the people to access large volumes of data and makes use of new applications. IoT in smart cities creates an intelligent environment and consuming more energy. IoT makes the people to participate in a smart environment which results of increasing productivity, efficiency, decreasing cost and improving the quality of life. IoT devices are intelligent devices which need memory space, continuous power supply, processing capability and power efficient communication protocols.

Many new IoT standards, protocols and products have been initiated. Wireless communications such as sensors and actuators are interconnected together that play the key role in IoT-based Smart City environment. However there are two challenges in smart cities using IoT devices. First challenge is all IoT devices share data across the platforms through cloud and establishing communication between IoT devices through a protocol is another challenge.

The IoT is enabled by the latest developments of radio frequency identification (RFID) and smart sensors. In IoT environments, smart devices communicate and collaborate directly, without the help of humans which deliver tremendous new applications. It makes the environment as smart to connect intelligent devices. Machine-to-machine (M2M) communication is done in the IoT platform where machines exchange information among them and monitor the environment activities. For example, if physical devices can monitor the traffic or pollution in a city without human interaction then that makes the city smart. When intelligent machines monitor the heating, ventilation and air conditioning in a home then that is smart home. IoT makes the machines to talk themselves, to share information themselves and to take decisions without the help of humans. It has a tremendous home and business applications which will improve the quality of life and to grow the economy of the world. IoT devices needs to be developed in order to satisfy the requirements of the customers in terms of availability- anywhere and anytime. Also new communication protocols have to be developed that can work in various environments.

1.1 Related works

Palattella et al. (2013) proposed a standardised protocol stack for the IoT. This protocol meets the important criteria like power efficiency (IEEE 802.15.4-2006 PHY layer), reliability (IEEE 802.15.4e MAC layer) and internet connectivity (IETF 6LoWPAN adaptation layer) among IoT devices. Sheng et al. (2013) described the importance of IoTs and current scenarios and research activities in the field of IoTs. The authors presented a survey on internet engineering task force (IETF) protocol suite by presenting the technical challenges and opportunities present in each layer. Bontu et al. (2014) introduced a wide area communication system based on the IoTs that works within the operator’s licensed macrocellular band. It is well suitable for low complexity IoT modules, low energy with low priority and irregular IoT traffic environment. The above authors also introduced a simplified protocol for IoT named air interface protocol and a simultaneous access channel for uplink (UL) that will be used for IoT communication. Rani et al. (2017) proposed a cross layer protocol for multimedia IoTs which is helpful for cross communication in between the multimedia applications.

IoTs gives a platform where users, computing devices are able to sense and communicate the information with each other. Granjal et al. (2015) presented a survey on the existing protocols in IoT and some security mechanisms for secure communication in between the IoT devices from a protocol stack perspective. Aijaz and Aghvami (2015) presented the emerging scenario in cognitive M2M communications. Near field communication (NFC) can be used for short range communications between intelligent devices. NFC is used in the electronic payment systems and various security protocols are introduced to provide secure communications. But NFC is vulnerable to two impersonation attacks. He et al. (2015) proposed a new method called pseudonym-based NFC protocol that avoids the impersonations attacks. Zhou et al. (2016) proposed a protocol named enhanced channel aware routing protocol which is used in internet of underwater things. This proposed method solves two problems addressed in traditional CARP like reusability of previously collected sensory data and selecting the most appropriate relay node at each time point. Al-Fuqaha et al. (2015) presented an overview of IoT issues, challenges, protocols, elements, architecture and applications.

Buratti et al. (2016) implemented IoTs by using software defined wireless networking (SDWN) along with ZigBee and IPv6 protocols. The above authors proved that SDWN is well suitable for IoT applications where IoT nodes are in a static position. But when the situation is dynamic that is when the IoT nodes are having mobility then ZigBee and low power wireless personal area networks (6LowPAN) are working better. Ray et al. (2016) proposed a secure object tracking protocol for ensuring the visibility and traceability of an IoT device along the travel path in IoT platform. This protocol is based on RFID and uses some cryptographic primitives. Qiu et al. (2016) proposed an efficient tree-based self-organising protocol (ETSP) for IoT sensor nodes present in IoT network. The proposed ETSP protocol is used to adjust the network topology dynamically to balance energy consumption and prolong network lifetime.

IoT devices send information among themselves. So it is mandatory to share the secured information in between the IoT devices. Park (2017) proposed a new security certificate issuing protocol called elliptic-curve Qu-Vanstone (ECQV) which provides secured communication in between the IoT devices.

Samaras et al. (2013) introduced a modified protocol stack of the device profile for web services that comply with the 6LoWPAN architecture. 6LoWPAN protocol is good for communication in between IoT devices in the IoT platform. Chen et al. (2014) proposed a protocol named enhanced group mobility protocol which is used to reduce the packet loss ratio, signalling cost and handoff delay in the 6LoWPAN protocol. The above authors reduce the number of control messages in order to reduce the handoff delay. In order to reduce the multicast cost and delay, Wang (2015a) proposed a multicast scheme to attain the L3 multicast with the L2 unicast. In most of the IPv6 addressing protocols, the IP address of a particular node is fixed and it will not be changed throughout the communication process. So the hackers can easily monitor the status of a particular node by tracking its IP address. For preventing the IP address attack, Wang and Mu (2015) proposed a method which changes the IP address dynamically without extra overheads. Wang (2015b) proposed a mobility frame for 6LoWPAN in order to reduce the handover latency and the communication latency. Because of this mobility frame, the IoT devices do not need to be configured with a care of address and can directly be accessed without visiting their home agent. Qiu et al. (2016) proposed a security framework scheme named as enhanced mutual authentication and key establishment scheme for the communication in between IoT devices in 6LoWPAN networks. This scheme makes the IoT device to make a secure authenticated communication with the remote server by using a session key established between them in order to prevent from the malicious attacks.

Vinoski (2006) proposed a new protocol called advanced message queuing protocol. IoT technologies are becoming the emerging field now. So more wireless technologies needed for making communication in between IoT devices. Chang (2014) gave an overview of the need Bluetooth technology in IoT platform. Harris et al. (2016) presented the primary issues by using Bluetooth technology in IoT environment. Mois et al. (2017) presented a comparative evaluation study of three different protocols like user datagram protocol (UDP), hypertext transfer protocol and Bluetooth technology. Bormann et al. (2012) proposed an application protocol which is based on constrained application protocol (CoAP) that can be applicable for billions of small internet nodes. IoT is a booming technology which is used in many applications including health care. To make secured transmission of the sensitive information, secure CoAP authorises the use of datagram transport layer security (DTLS). Raza et al. (2013) proposed Lithe which is an integration of DTLS and CoAP for the IoT devices. The authors also proposed a header compression scheme and the results of the proposed scheme gains in terms of processing time, energy consumption and packet size.

Correia et al. (2016) proposed a new holistic framework for the planning of registration steps in CoAP/observer-based wireless sensor networks which are able to reduce the energy consumption and average load in the M2M nodes. In IoT, there is a high demand for home automation systems, time synchronisation techniques for low power sensor modules. Son et al. (2016) proposed a lightweight time synchronisation algorithm for CoAP-based home automation system networks and produces an average error of 1 ms and a network overhead reduction of 17%. Betzler et al. (2016) proposed a new congestion control mechanism algorithm for CoAP called as CoCoA. In this algorithm, a novel round-trip time estimation scheme is introduced which is able to give dynamic and controlled retransmission which is more suitable for IoT communications.

Babovic et al. (2016) evaluated IoT communication and messaging protocols for analysing the performance of web. Routing protocol for low power and lossy networks (RPL) outperforms well in terms of quality of service (QoS), device management and energy saving performance. Le et al. (2013) analysed several different types of internal threats that are aimed at rank property in RPL protocol. Kim et al. (2017) investigated the load balancing and congestion problems present in RPL and introduced an effective queue-based approach which reduces the above problems. Si et al. (2013) proposed architecture for intelligent server management, which is based on extensible messaging and presence protocol (XMPP). This framework simplifies the server management and increase flexibility and scalability. Guo et al. (2015) proposed a new security mechanism that satisfies most of the cryptography principles for the XMPP-based communications. Buratti et al. (2016) tested different protocols in the IoT platform by taking several performance metrics. de Almeida Oliveira and Godoy (2016) presented a feasibility analysis of the ZigBee protocol of the wireless dynamic sensor network.

1.2 Motivation and justification of the proposed work

The growth of digital technologies makes the world with lots of sophistication in terms of accessing the information from various sources in an efficient, flexible way. In most of the developing countries, information is available in and around them, but not in the manner, it could be easily accessible. For example, if a city is affected through a heavy cyclone which results in a lot of damage includes road block, power failure, mobile phone signal issues, transport stumble, etc. Local lads have the information to solve the above issues temporarily but tourist found difficult to cope with the situation. It is necessary to use some advanced technologies to collect the above mentioned critical information from different local sources and store it in a distributed point of contact. IoT provides the technology which can be used to collect the required sensitive information from various critical areas of the city and if so, it is termed as Smart City. Success of

Smart City is based on how efficiently IoT has been deployed with various communication technologies to collect, store and distribute the information when required.

IoT protocols play a vital role in creating a distributed environment so that all the devices within the network can interact with each other. These protocols are designed in such a way it can sense information in a constrained environment and transform the same to the Internet through supporting protocols. Some of its protocols support instant messaging (IM), multi-party chat, voice and video calls, etc. to enable real world communication. M2M communication and machine to person communication are other features of IoT protocols which makes the implementation of the Smart City concept feasible with the help of IoT.

1.3 Organisation of the paper

The rest of the paper is organised as follows. Section 2 describes the IoT protocols for smart environments. Experimental design is given in Section 3. IoT protocols are analysed in Section 4. Finally the paper concludes in Section 5.

2 IoT protocols

2.1 6LowPAN

6LoWPAN (Kushalnagar et al., 2007) is the specification of mapping services needed by IPv6 (internet protocol) over 6LoWPAN in order to maintain IPv6 network. The main key parameters provided by 6LoWPAN are reduction in transmission overhead, header compression and fragmentation in order to attain the IPv6 maximum transmission unit requirement. 6LoWPAN was developed by IETF for the development of IoT devices and M2M applications. 6LoWPAN consists of IoT devices with limited in energy, memory, throughput and power. The IoT devices can communicate with each other through a low-power wireless standard. The IoT devices should be adjusted to low power, low bandwidth and low cost network communicating over IEEE 802.15.4 standard. 6LoWPAN supports header compression, which reduces the transmission overhead. It follows a header which has four types. 00 means no 6LoWPAN header is present so that packets will be discarded that do not accord to the 6LoWPAN. One represents a dispatch header which is used to perform compression of IPv6 header, ten represents a mesh addressing which identifies IEEE 802.15.4 packets which have to be sent to the link layer. And 11 represent's fragmentation which is used to remove a lot of IPv6 overheads.

2.2 MQTT

Message queue telemetry transport (MQTT) (Locke, 2010) is designed to provide a connection between the application and middle wares on one side and on the other side it will be used for providing communication between IoT devices. It is also designed for making remote connections. MQTT attains limited bandwidth and small code footprint. The main components of MQTT are publishers, subscribers and brokers. The methods in MQTT are connect, disconnect, subscribe, unsubscribe and publish. The connect method helps to connect the sensor and disconnect method helps to disconnect the sensor and subscribe method helps to subscribe the services and unsubscribe method helps to unsubscribe the services and publish method helps to publish the data from the different sensors.

For example, temperature sensor using MQTT publishes the information to broker and get subscription from laptops and mobile devices. SMQTT means secure MQTT and it is an extension of MQTT. MQTT is suitable for resources of devices which has low bandwidth links. This is placed on the top of the transmission control protocol (TCP) protocol to deliver messages. The major specifications in MQTT are connection semantics, routing and end point which is used to give a QoS.

By using the three components subscriber, publisher and broker of MQTT, an authorised device need to register like subscriber and obtains the information. And it has to inform the broker before publishers publish that data. Here publisher acts as a generator of the data. In the second step, publisher transmits or passes the information to the subscribers with the help of brokers. Finally, a broker is used to attain the security by checking with an authority of publishers and subscribers.

MQTT makes the connection in between IoT devices with applications and middleware. MQTT is an optimal connection protocol for IoT device which uses one-to-one, one-to-many and many-to-many routing mechanisms. MQTT uses the publish/subscribe pattern which is shown in Figure 1. As shown in Figure 1, it consists of subscriber, broker and publisher. MQTT presented on the top of the TCP protocol, which is suitable for resource constrained IoT devices. If an IoT device wants to communicate with other device then it will be registered as a subscriber and a publisher IoT device delivers the required data through the broker. MQTT is used by many applications such as agriculture monitoring, healthcare and social network notifications. It is an ideal messaging protocol for the IoT devices by taking the minimum requirements like bandwidth, code footprint, power consumption and message data overhead. Three levels of QoS is used by MQTT. Figure 2 shows the packet format of MQTT.

Figure 1 MQTT architecture

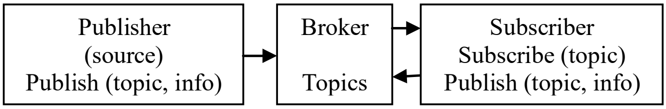


Figure 2 MQTT packet format

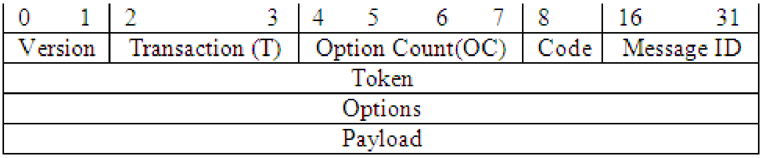
0 1 2 3	4	5 6	7
Message type	UDP	QoS level	Retain
Remaining length (1–4 bytes)			
Variable length header			
Variable length message payload			

2.3 Constrained application protocol

This CoAP protocol (Shelby et al., 2013; Bormann et al., 2012) was developed by IETF constrained restful [representational state transfer (REST)] environments. IoT applications are used this protocol as an application layer protocol. CoAP is an application layer protocol which can be used for many IoT applications. It makes communication among machines by providing request-response transactions. A web transfer protocol which is based on REST is defined by CoAP on top of hypertext transfer protocol. REST works on stateless client-server architecture as a cacheable connection protocol. Many social network applications use REST and it enables clients and servers to use web services. CoAP satisfies the demands of IoT devices like low power consumption, communication capabilities and operation in the existence of lossy and noisy links. CoAP was designed for M2M communications which is applicable for smart energy and building automation. CoAP is processed based on request-response operation between IoT nodes. Low power sensors which have power constraint issue can be able to use CoAP. CoAP protocol was only built over UDP instead of TCP. There are two sub-layers present in the CoAP. Messaging sub-layer is responsible to identify duplications and gives reliable communications. Request-response sub-layer handles the communications among IoT devices.

CoAP can be used to provide unicast transmission in many IoT applications because it is bound with UDP. CoAP uses four types of messages like confirmable, non-confirmable, reset and acknowledgement. In confirmable mode, an IoT device has to wait for a specific time for getting reply from another IoT device. In non-confirmable mode, an IoT device sends a message to another IoT device without waiting for the acknowledgement message. Reset message is sent when data is lost. CoAP uses methods such as get, put, post and delete to inquire about another IoT device status. For example, a server uses get method to get the current temperature from a temperature sensor. Figure 3 shows the message format of CoAP. CoAP handles the congetion control itself and the main features of CoAP are resource observation, block-wise resource transport, resource discovery and security.

Figure 3 CoAP message format



0 1	2 3	4 5 6 7	8	16 31
Version	Transaction (T)	Option Count(OC)	Code	Message ID
Token				
Options				
Payload				

2.4 *Extensible messaging and presence protocol*

XMPP (Saint-Andre, 2011) is an IETF IM which aims for IM in between IoT devices for multiuser chatting, voice and video calling. It is a messaging protocol which was developed to support an open, secure, spam free and decentralised messaging protocol. XMPP is a platform independent which is able to allow communicating with each other by sending IM over the internet. XMPP satisfies the cryptography principles in between the IoT devices like authentication, access control and end-to-end encryption. The gateways can be present in between foreign messaging networks which illustrate the overall behaviour of XMPP protocol and it allows the IM application within the scope of IoT.

Internet-based platforms in a decentralised fashion can be used to run the XMPP protocol. By allowing XMPP protocol, the addition of new applications on top of core protocols and make them secured can be done. XMPP works within the requirements of the IoT devices and works over a various internet-based platforms. Using extensible markup language (XML) stanzas, XMPP connects an IoT device with another IoT device. XML stanzas contain a piece of code which contains message, presence and information/query from the IoT device. Message is used to find the client IoT device and the server IoT device by using the address. The presence is used to identify whether the IoT device is an authorised one or not. The information/query stanza is used to pair the message between the IoT devices. XMPP also gives the information about the network and the availability of the IoT devices. It is able to carry a wide variety of payloads for multimedia applications and web services. In XMPP, when an IoT device sends an information to another IoT device which is present in different domain, the client IoT device connects to the domain server and then directly connects to the another domain server without any intermediates.

2.5 Routing protocol for low power and lossy networks

The IETF routing protocol over low-power and lossy links (RPL) was introduced (Vasseur, 2011; Winter, 2012) as a link-independent routing protocol based on IPv6 for resource-constrained nodes called as RPL. This protocol is applicable for routing in long and low power networks. It maintains routing topology using low rate beaconing and beaconing rate is increased by detecting inconsistencies. Beaconing rate increases due to node failure or link failure. This will be avoided by including the routing information in the datagram itself. There are two types of routing in RPL. They are proactive and reactive. Proactive refers to maintain routing topology and reactive revolves routing inconsistencies.

The unique characteristics of low power and lossy networks (LLNs) provide several design choices for RPL. RPL routing performance is dependent on the routing environments. If convergence time is of utmost importance for a routing protocol operating in a ‘classic’ IP networks, the prime performance objective in LLNs is scalability and stability, which are the main challenges in LLNs where links and nodes are potentially highly unstable. Another objective is to bind the control traffic to save bandwidth and energy – the control traffic is negligible compared to the traffic that is considerably lower in ‘typical’ IP networks. RPL supports redistribution of routes. The route redistribution happens at the root, which will redistribute routes into other routing protocol that are directly connected as well as routes learnt from decentralised autonomous organisation (DAO) messages. Routes learnt from other routing protocols will not be redistributed into the RPL network domain. The root updates the application virtual routing and forwarding (VRF) with the routing update information. By using the redistribute RPL command, redistribution can be specified in an RPL network.

RPL separates the packet processing and forwarding from the routing optimisation objective that helps in low power long network. It supports integrity and confidentiality and also includes data-path validation and detects loops. The overall objectives of RPL are minimising energy, minimising latency and satisfying constraints. This operation requires bidirectional flow of communication. In some of the LLN scenarios, those links may exhibit asymmetric properties.

RPL was introduced to form a robust topology over lossy links for making communications in between the IoT devices by considering the minimal routing requirements. RPL supports simple and complex traffic models. A destination oriented directed acyclic graph (DODAG) is a directed acyclic graph which is used by RPL which has all the IoT devices. This graph has a root and children where all the IoT devices are arranged as nodes. Each IoT device in the node knows about its parents but they do not have information about related children nodes. RPL has minimum one path for each IoT device to the root to increase the performance and faster access. RPL has four types of control messages which are information object, destination advertisement object, DODAG information solicitation message and acknowledgment message. The destination advertisement object has the path information between the root and destination. The destination advertisement object has the information regarding the upward and downward traffic. DODAG information solicitation message is used by an IoT device to get DODAG information object message from its adjacent IoT device node. The acknowledgement message gives response to a destination advertisement object message which is sent by an IoT device. Initially, only root IoT device is present in a DODAG. The root IoT device starts to send its current location to all LLN and at each level the IoT devices register the parent path and participation path for its node. The registered IoT devices send their DODAG information object message and the whole graph is gradually built. RPL operates in non-sorting mode and sorting mode. One of the parent in the DODAG graph is called as a preferred parents and it is used for routing toward the root. The IoT devices present in DODAG send signalling information in order to construct and maintain the graph. The root IoT device sends a DODAG information object message to its neighbour IoT device to announce a minimum rank value. After receiving this message, an IoT device present in a node will update the list of its neighbours, calculate its own rank value, choose its preferred parent and start transmitting messages which has its own rank. When an IoT device receives a destination advertisement object message, it will update its routing table. An adaptive timer mechanism is used by RPL to control the sending rate of messages. This timer mechanism is called as trickle timer which verifies if an IoT device has expiry routing information. The frequency of the messages in between the IoT devices depends on the stationary of the network. RPL supports many applications like smart home, Smart City and smart industry environments.

2.6 ZigBee protocol

ZigBee (2008) is highly established by IEEE 802.15.4. The ZigBee protocol is explained by the network layer 3. This protocol highly works with two network layers i.e., 1 and 2. For explaining or defining the additional communication enhancements it uses 3 and 4 layers. These enhancements involve the communication with the useful nodes, encryption for security, and for data routing and enables mesh network. Mostly the ZigBee protocol is wireless sensor network and uses the mesh topology highly. Mesh topology can be formed with the help of ZigBee protocol. The components in the ZigBee protocol are ZigBee device object (ZDO) and application support sub-layer (APS). The first component ZDO manages the Device management, security and policies. The second component APS acts as an interfacing and control server. It also acts as a bridge between network and other layers.

Table 1 Characteristic of IoT protocols

	Layers		*Protocols*	*Security*	*Power consumption*	*Transport layer*	*Architecture*	*Header size (bytes)*
Application protocol	Application layer		CoAP	DTLS	Low	UDP	Request/response	4
	Session (messaging protocol)		MQTT	TLS/SSL	Low	TCP	Publish/subscribe	2
	Session		XMPP	TLS/SASL	Low	TCP	Both	-
Infrastructure protocol	Network layer	Routing protocol	RPL	AES-128	Low	TCP	Request/response	15
		Encapsulation	6LoWPAN	SSL	Low	TCP/UDP	Request/response	15
	Data link layer		Zigbee	SSL/TLS	Low	TCP	Request/response	15

The ZigBee topologies are star topology, cluster tree topology and mesh topology. The star topology consists of a coordinator node and age devices. All the age devices are connected with the one coordinator node. This topology creates a simple local area network. The second topology is the cluster tree topology and it consists of different clusters that connect many routers and coordinator node. This router forms a tree like structure with a coordinator node. This is why it is called as cluster tree. In mesh topology the mesh network forms with different routers. Routers act as a backbone of network. The different types of routers are attached to the end devices. At one end of mesh network, a coordinator node acts as a gate way. From here it offers connectivity such as internet.

In ZigBee mesh, one node can communicate with another node in its range. If nodes are not in range the messages are transferred through the intermediate nodes. This allows the network movement over large areas. ZigBee has components like ZigBee coordinator, ZigBee router and ZigBee end device. The ZigBee coordinator forms the root of ZigBee network and this network tree act as a bridge between two networks. There will be one ZigBee coordinator in each network which initiates the network. It stores information about the network. It acts as a trust centre and repository for security keys. The ZigBee Router is capable for running applications as well as transferring information between nodes connected to it. The ZigBee end device needs very low memory requirements and cost is also very low. The main applications of ZigBee protocol are home monitoring, smart phone, health care, remote control and telecom services.

2.7 Characteristic comparison of protocols

Table 1 summarises the characteristics of IoT protocols of studied in this paper. The characteristics are presented in terms of layers, security, power consumption, transport layer, architecture and header size. CoAP, MQTT and XMPP are application protocols which lie in application layer and session layer. These protocols consume low power consumption because of modifying some HTTP functionalities based on REST. RPL, 6LoWPAN and Zigbee protocols support minimal routing requirements by constructing a robust topology over lossy networks. So it also consumes low power. All protocols analysed in this paper are providing security because it is constructed based on DTLS, transport layer security (TLS), secure socket layer (SSL) and simple authentication and security layer (SASL). RPL uses advanced encryption standard (AES) mechanism. QoS is also given because of integrity and confidentiality of message exchange. But QoS is not given by XMPP and ZigBee protocols because these protocols offer low data rate services. IoT protocols stay on security protocols at transport layer and serve any upper layer including all the application protocols that reply on TCP/UDP. IoT protocols follow the particular architecture model which is shown in Table 1.

3 Experimental setup

In the simulation experiment, nodes were placed uniformly in an area of 500 m × 500 m in a rectangular field. The traffic is constant bit rate (CBR) with 250 bytes data packet. The simulation scenarios are created by the setdest tool of ns-3. The channel rate is 128 Kbps and transmission range is 100 m for all nodes. The simulation parameters are summarised in Table 2. In order to analyse the protocols, the QoS parameters such as end-to-end delay, packet delivery ratio (PDR), throughput and reliability are taken.

Table 2 Simulation parameters

Parameters	*Particulars*
Simulator	Network simulator-3
Protocol	6LoWPAN, MQTT, CoAP, XMPP, ZigBee, RPL
No. of nodes	10 nodes
Simulation area	500 m × 500 m
Node movement	Static
Frequency band	2,400–2,483.5 MHz
Channel rate	250 kbps
Traffic	CBR
CBR packet size	250 bytes
Payload size	10 byte
MAC header	27 byte
Transmission range	100 m
Maximum packet size	1,024 bytes

4 Performance evaluation

4.1 Performance metrics

4.1.1 End-to-end delay

The end-to-end delay is defined as the interval that elapses between the time a packet is sent and the time at which the packet is successfully delivered (Rahman et al., 2011).

$$Delay = \frac{1}{R}\sum_{j=1}^{n}\left(r_j - s_j\right) \tag{1}$$

where R is the number of successfully received packets, j is unique packet identifier, r_j is time at which a packet with unique id j is received, s_j is time at which a packet with unique id j is sent and delay is measured in sec. It should be less for high performance.

4.1.2 Packet delivery ratio

PDR is the ratio of the number of data packets delivered to the destination to the number of packets generated by the source node (Anisi, 2013) as below:

$$PDR = \frac{TP_{RD}}{TP_s} \tag{2}$$

where

$$
\begin{aligned}
&TPRD\,(total\ number\ of\ \ packets\ received\ at\ destination)\\
&= \sum_{i=1}^{n} N_s\,(Number\ of\ source\ nodes)\times N_p\,(Number\ of\ \ packets)\\
&\times PR_n\,(Received\ at\ destination\ by\ each\ ode)
\end{aligned}
\tag{3}
$$

$$
\begin{aligned}
TP_s\,(total\ \ packet\ sent) = \sum\nolimits_{i=1}^{n} &Nn(Number\ of\ \ nodes)\\
&\times Np_{ni}\,(Number\ of\ \ packets)
\end{aligned}
\tag{4}
$$

4.1.3 Throughput

Throughput can be defined as the number of data packets generated by source node to the number of data packets received in the destination node.

$$Throughput = \frac{N_{RD}\times 8}{T_s\times 1,000}kbps \tag{5}$$

where N_{RD} is number of bytes received and T_s is simulation time.

4.1.4 Reliability

Reliability is defined as the successful end-to-end data delivery ratio (Khandani et al., 2005).

$$Reliability\left(r_0, r_1, \ldots, r_{h-1}, r_h\right) = \exp\left(-\sum_{i=1}^{h}\frac{d_{ri-1ri}^{k}}{snr_{ri-1ri}}\right) \tag{6}$$

where

$(r_0, r_1, \ldots, r_{h-1}, r_h)$	is route
d_{ri-1ri}^{k}	is distance between the nodes
snr_{ri-1ri}	is the transmitted signal-to-noise power.

4.2 Performance analysis

In this section, simulation results for the selected six IoT protocols for the performance metrics of end-to-end delay, PDR, throughput and reliability are elaborated. Each protocol is simulated and analysed under two scenarios such as by varying the traffic load from 5 to 50 pkts/s and by varying number of nodes from 10 to 100 nodes for fixed minimum speed of 0 m/s.

4.2.1 Scenario: varying the traffic load

In this scenario, the following graphs show that performance comparison between selected six IoT protocols separately. Figure 4 shows the graphs for end-to-end delay (sec) versus traffic load (pkts/s). Figure 4 shows the MQTT exhibits lesser values of end-to-end delay, because it is optimal connection protocol, therefore it shows a better delay performance than the other IoT protocols at low pause time, when increasing the traffic load. The presence of additional messages in advance leads to lower end-to-end delay in 6LoWPAN.

Figure 4 End-to-end delay(s) vs. traffic load (pkts/s) (see online version for colours)

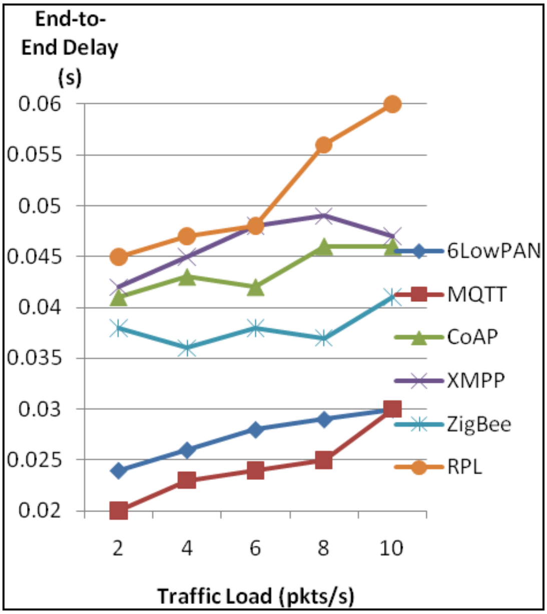


Figure 5 PDR (%) vs. traffic load (pkts/s) (see online version for colours)

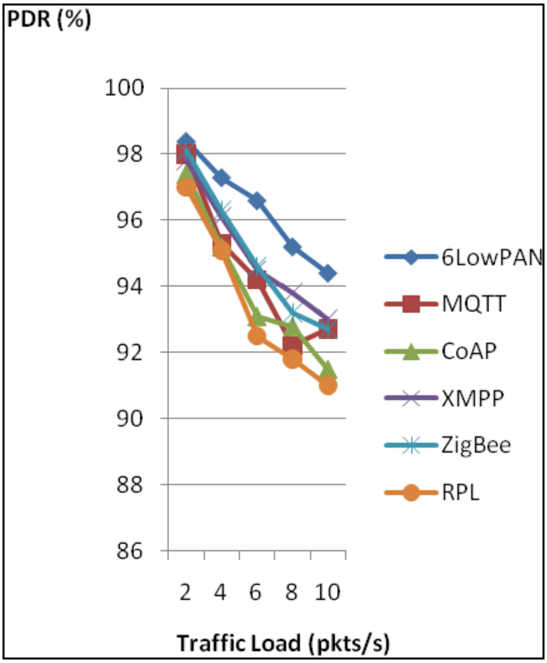


Figure 5 shows packet delivery ratio (%) versus traffic load. Based on the simulation results shown in Figure 5, the packet delivery ratio of 6LoWPAN is higher (98.4%) than other IoT protocol when the traffic load is minimum (2 pkts/s) and packet delivery ratio is gradually decreases when increasing the traffic load from 2 to 10 pkts/s. ZigBee and XMPP are better suited to constrained environments, but 6LoWPAN is dynamically on-demand IoT protocol that means it can be adjusted dynamically and send data better than other IoT protocols, ZigBee shows better (98%) packet delivery ratio than other IoT protocol. RPL shows worst performance than CoAP, MQTT and XMPP protocols because it is lossy networks.

Figure 6 Throughput (kbps) vs. traffic load (pkts/s) (see online version for colours)

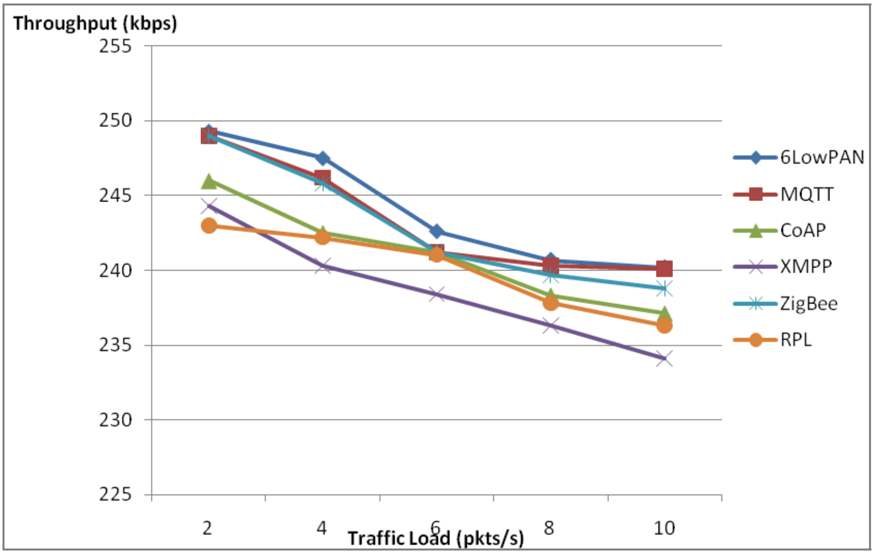
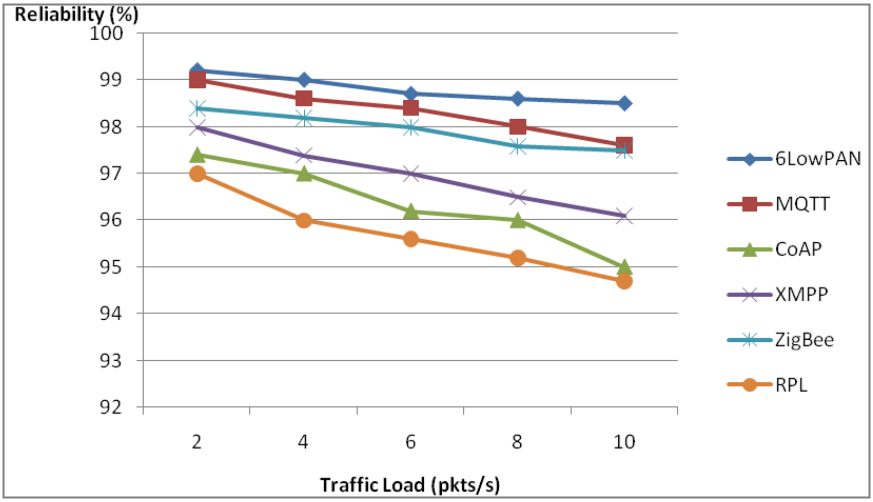


Figure 7 Reliability (%) vs. traffic load (pkts/s) (see online version for colours)



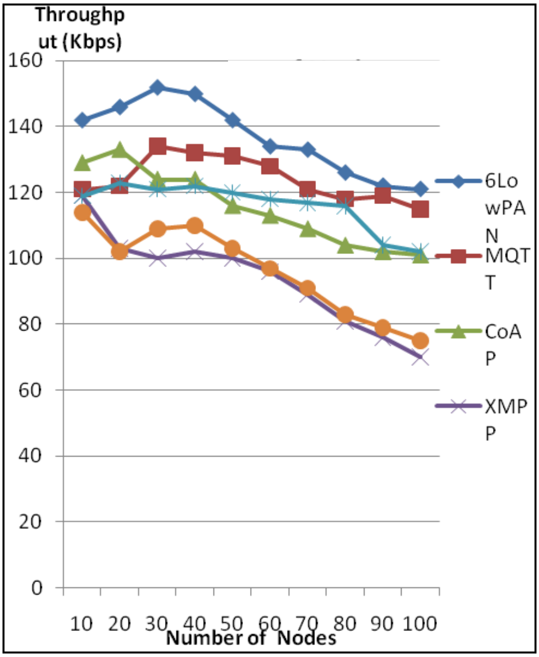
In Figure 6 simulation results of throughput (kbps) versus traffic load (pkts/s) are plotted. From the Figure 6 it is observed that, on increasing the traffic load, 6LoWPAN provides higher throughput than other IoT protocols, which delivers data packets at higher rate due to their operations which is on-demand in nature. RPL has worst performance in throughput than other IoT protocols because most of the nodes (devices) are not participate in data transfer. Another reason is link breakage since RPL cannot repair route of breakage path. ZigBee and MQTT show better performance than CoAP, XMPP and RPL but less than 6LoWPAN.

From Figure 7, it is observed that the performance of 6LowPAN protocol is very good with low load in comparison to MQTT, ZigBee and XMPP. Performance of protocols degrades slowly due to overhead of the control packets transmitted in the group and a larger number of packets are lost due to collisions. RPL and CoAP provide worst performance for reliable network.

4.2.2 Scenario: varying the number of nodes

In this scenario, the following graphs show that performance comparison between selected six IoT protocols separately.

Figure 8 Throughput (kbps) vs. number of nodes (see online version for colours)



In Figure 8 simulation results of throughput (kbps) versus number of nodes are plotted. From Figure 8 it is observed that, on increasing the number of nodes, 6LowPAN provides higher throughput than other protocol. 6LowPAN delivers data packets at higher rate due to their operations in nature. XMPP has worst performance in throughput than other protocols because most of the nodes are not participate in data transfer. MQTT and ZigBee show better performance than CoAP, RPL and XMPP but less than 6LowPAN.

Figure 9 shows the graphs for end-to-end delay (sec) versus number of nodes. Figure 9 shows the 6LowPAN exhibits lesser values of end-to-end delay, because its route discovery mechanism is fast, therefore 6LowPAN shows a better delay performance than the other protocols at low pause time when increasing the number of nodes. XMPP and RPL provides worst performance by delivering data packets with high delay.

Based on the simulation results shown in Figure 10, the packet delivery ratio of 6LowPAN is higher (99%) than other protocol when the number of node is minimum (ten nodes) and packet delivery ratio is gradually decreases when increasing the number of nodes from 10 to 100 nodes. MQTT shows better (98%) packet delivery ratio than ZigBee protocol. RPL shows worst performance than CoAP and XMPP.

Figure 9 End-to-end delay(s) vs. number of nodes (see online version for colours)

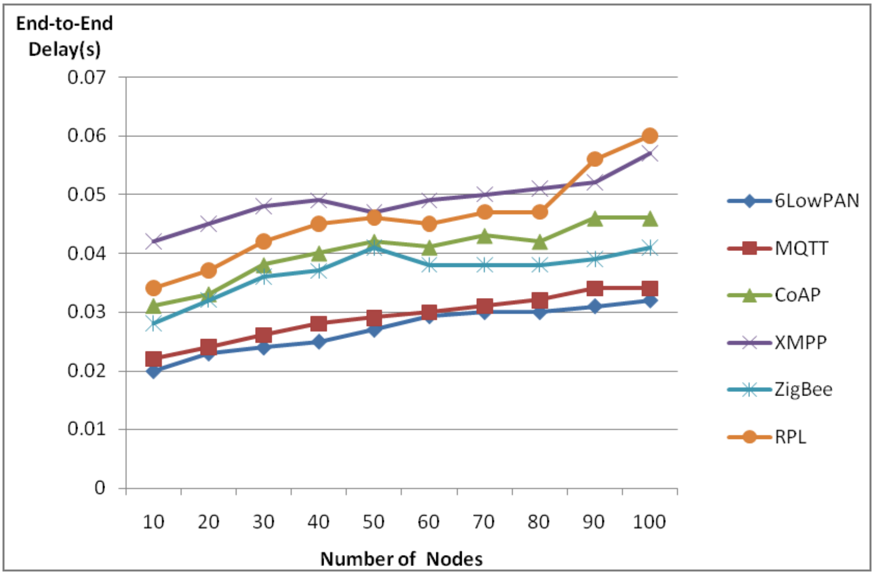


Figure 10 PDR (%) vs. number of nodes (see online version for colours)

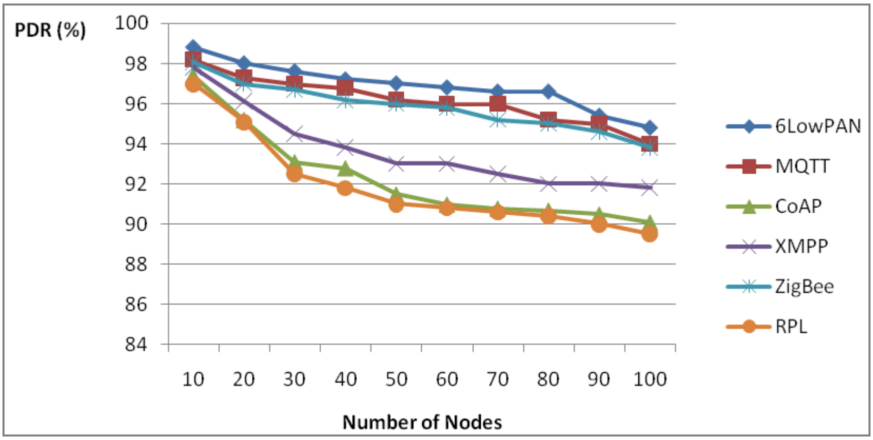
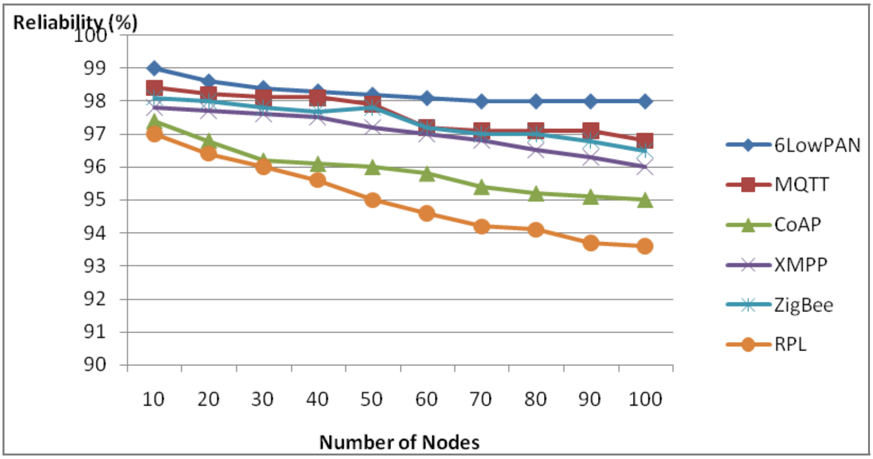


Figure 11 shows the reliability versus number of nodes with low load, 6LowPAN protocol achieves high reliability because collisions between messages are very rare under low load, Nevertheless, MQTT and ZigBee achieve the highest reliability among the other protocol. It is observed that, on increasing the number of nodes. As the network becomes strongly connected, the reliability also improves. CoAP and XMPP offer average reliability. RPL has less reliability, because of its high control overhead.

Figure 11 Reliability (%) vs. number of nodes (see online version for colours)



5 Conclusions

In this paper, various IoT protocols that will be applicable to smart environments are taken and are evaluated to identify the best protocol which is suitable for IoT smart environments. The protocols involved in this paper are 6LoWPAN, MQTT, CoAP, XMPP, RPL and ZigBee protocol. The simulation results IoT protocols in smart environment are presented and it is evaluated based on the performance metrics like end-to-end delay, packet delivery ratio, throughput and reliability. Finally, it is concluded that 6LowPAN, MQTT and ZigBee protocols are best suited for the millions of tiny internet nodes.

References

Aijaz, A. and Aghvami, A.H. (2015) ‘Cognitive machine-to-machine communications for internet-of-things: a protocol stack perspective’, *IEEE Internet of Things Journal*, Vol. 2, No. 2, pp.103–112.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M. (2015) ‘Internet of things: a survey on enabling technologies, protocols, and applications’, *IEEE Communication Surveys & Tutorials*, Vol. 17, No. 4, pp.2347–2376.

Anisi, M.H., Abdullah, A.H. and Razak, S.A. (2013) ‘Energy efficient and reliable data delivery in wireless sensor networks’, *Wireless Networks*, May, Vol. 19, No. 4, pp.495–505.

Babovic, Z.B., Protic, J. and Mlutinovic, V. (2016) ‘Web performance evaluation for internet of things applications’, *IEEE Translations and Content Mining*, Vol. 4, No. 1, pp.6974–6992.

Betzler, A., Gomez, C., Demirkol, I. and Paradells, J. (2016) 'CoAP congestion control for the internet of things', *IEEE Communications Magazine*, July, Vol. 54, No. 7, pp.154–160.

Bontu, C.S., Periyalwar, S. and Pecen, M. (2014) 'Wireless wide-area networks for internet of things', *IEEE Vehicular Technology Magazine*, March, Vol. 9, No. 1, pp.54–63.

Bormann, C., Castellani, A.P. and Shelby, Z. (2012) 'CoAP: an application protocol for billions of tiny internet nodes', *IEEE Internet Comput.*, April, Vol. 16, No. 2, pp.62–67.

Bormann, C., Castellani, A.P. and Shelby, Z. (2012) 'CoAP: an application protocol for billions of tiny internet nodes', *IEEE Internet Comput.*, April, Vol. 16, No. 2, pp.62–67.

Buratti, C., Stajkic, A., Gardasevic, G., Milardo, S., Abrignani, M.D., Mijovic, S., Morabito, G. and Verdone, R. (2016) 'Testing protocols for the internet of things on the EuWIn platform', *IEEE Internet of Things Journal*, February, Vol. 3, No. 1, pp.124–133.

Chang, K-H. (2014) 'Bluetooth: a viable solution for IoT?', *IEEE Wireless Communications*, December, Vol. 21, No. 6, pp.6–7.

Chen, Y-S., Hsu, C-S. and Lee, H-K. (2014) 'An enhanced group mobility protocol for 6LoWPAN-based wireless body area networks', *IEEE Sensors Journal,* March, Vol. 14, No. 3, pp.797–807.

Correia, N., Sacramento, D. and Schütz, G. (2016) 'Dynamic aggregation and scheduling in CoAP/observe-based wireless sensor networks', *IEEE Internet of Things Journal*, December, Vol. 3, No. 6, pp.923–936.

de Almeida Oliveira, T. and Godoy, E.P. (2016) 'ZigBee wireless dynamic sensor networks: feasibility analysis and implementation guide', *IEEE Sensors Journal*, June, Vol. 16, No. 11, pp.4614–4621.

Granjal, J., Monteiro, E. and Sa Silva, J. (2015) 'Security for the internet of things: a survey of existing protocols and open research issues', *IEEE Communication Surveys and Tutorials*, Third Quarter, Vol. 17, No. 3, pp.2577–2586.

Guo, L., Wu, J., Xia, Z. and Li, J. (2015) 'Proposed security mechanism for XMPP-based communications of ISO/IEC/IEEE 21451 sensor networks', *IEEE Sensors Journal*, May, Vol. 15, No. 5.

Harris, A.F., Khanna, V., Tuncay, G., Want, R. and Kravets, R. (2016) 'Bluetooth low energy in dense IoT environments', *IEEE Communications Magazine*, December, Vol. 54, No. 12, pp.30–36.

He, D., Kumar, N. and Lee, J-H. (2015) 'Secure pseudonym-based near field communication protocol for the consumer internet of things', *IEEE Transactions on Consumer Electronics*, February, Vol. 61, No. 1, pp.56–62.

Khandani, A., Modiano, E., Abounadi, J. and Zheng, L. (2005) 'Reliability and route diversity in wireless networks', *Proc. Conference on Inform. Science and System*.

Kim, H-S., Kim, H., Paek, J. and Bahk, S. (2017) 'Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks', *IEEE Transactions on Mobile Computing*, April, Vol. 16, No. 4, pp.964–979.

Kushalnagar, N., Montenegro, G. and Schumacher, C.P. (2007) 'IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals', *I.R. 4919*, August.

Le, A., Loo, J., Lasebae, A., Vinel, A., Chen, Y. and Chai, M. (2013) 'The impact of rank attack on network topology of routing protocol for low-power and lossy networks', *IEEE Sensors Journal*, Vol. 13, No. 10, pp.3685–3692.

Locke, D. (2010) *MQ Telemetry Transport (MQTT) v3*, 1 Protocol Specification, IBM Developer Works, Tech. Lib., Markham, ON, Canada.

Mois, G., Folea, S. and Sanislav, T. (2017) 'Analysis of three IoT-based wireless sensors for environmental monitoring', *IEEE Transactions on Instrumentation and Measurement*, August, Vol. 66, No. 8, pp.2056–2064.

Palattella, M.R., Accettura, N., Vilajosana, X., Watteyne, T., Grieco, L.A., Boggia, G. and Dohler, M. (2013) 'Standardized protocol stack for the internet of (important) things', *IEEE Communications Surveys & Tutorials*, Third Quarter, Vol. 15, No. 3, pp.1389–1406.

Park, C-S. (2017) 'A secure and efficient ECQV implicit certificate issuance protocol for the internet of things applications', *IEEE Sensors Journal*, April, Vol. 17, No. 7, pp.2215–2223.

Qiu, T., Liu, X., Feng, L., Zhou, Y. and Zheng, K. (2016) 'An efficient tree based self organizing protocol for internet of things', *IEEE Translations and Content Mining*, Vol. 4, No. 1, pp.3535–3546.

Rahman, R.A., Kassim, M., Yahaya, C.K.H.C.K. and Ismail, M. (2011) 'Performance analysis of routing protocol in WiMAX network', *IEEE ICSET*, June, pp.153–157.

Rani, S., Ahmed, S.H., Talwar, R., Malhotra, J. and Song, H. (2017) 'IoMT: a reliable cross layer protocol for internet of multimedia things', *IEEE Internet of Things Journal*, June, Vol. 4, No. 3, pp.832–839.

Ray, B.R., Chowdhury, M.U. and Abawajy, J.H. (2016) 'Secure object tracking protocol for the internet of things', *IEEE Internet of Things Journal*, Vol. 3, No. 4, pp.544–553.

Raza, S., Shafagh, H., Hewage, K., Hummen, R. and Voigt, T. (2013) 'Lithe: lightweight secure CoAP for the internet of things', *IEEE Sensors Journal*, October, Vol. 13, No. 10, pp.3711–3720.

Saint-Andre, P. (2011) *Extensible Messaging and Presence Protocol (XMPP): Core*, Internet Eng. Task Force (IETF), Request for Comments: 6120, Fremont, CA, USA.

Samaras, I.K., Hassapis, G.D. and Gialelis, J.V. (2013) 'A modified DPWS protocol stack for 6LoWPAN-based wireless sensor networks', *IEEE Transactions on Industrial Informatics*, February, Vol. 9, No. 1, pp.209–217.

Shelby, Z., Hartke, K., Bormann, C. and Frank, B. (2013) *Constrained Application Protocol (CoAP), draft-ietf-core-coap-18*, Internet Eng. Task Force (IETF), Fremont, CA, USA.

Sheng, Z., Yang, S., Yu, Y., Asilakos, A.V., Mccann, J.A. and Leung, K.K. (2013) 'A survey on the IETF protocol suite for the internet of things: standards, challenges and opportunities', *IEEE Wireless Communications*, December, Vol. 20, No. 6, pp.91–98.

Si, P., Song, C. and Zhou, X. (2013) 'Intelligent server management framework over extensible messaging and presence protocol', *China Communications*, May, Vol. 10, No. 5, pp.128–136.

Son, S-C., Kim, N-W., Lee, B-T., Cho, C.H. and Chong, J.W. (2016) 'A time synchronization technique for CoAP-based home automation systems', *IEEE Transactions on Consumer Electronics*, February, Vol. 62, No. 1, pp.10–16.

Vasseur et al. (2011) *RPL: The IP Routing Protocol Designed for Low Power and Lossy Networks*, Internet Protocol for Smart Objects (IPSO) Alliance, San Jose, CA, USA.

Vinoski, S. (2006) 'Advanced message queuing protocol', *IEEE Internet Computing*, December, Vol. 10, No. 6, pp.87–89.

Wang, X. (2015a) 'Multicast for 6LoWPAN wireless sensor networks', *IEEE Sensors Journal*, May, Vol. 15, No. 5, pp.3076–3083.

Wang, X. (2015b) 'A mobility frame for 6LoWPAN WSN', *IEEE Sensors Journal*, April, Vol. 16, No. 8, pp. pp.2755–2762.

Wang, X. and Mu, Y. (2015) 'Addressing and privacy support for 6LoWPAN', *IEEE Sensors Journal*, September, Vol. 15, No. 9, pp.5193–5201.

Winter, T et al. (2012) *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, Internet Eng. Task Force (IETF), Request for Comments: 6550 Fremont, CA, USA.

Zhou, Z., Yao, B., Xing, R., Shu, L. and Bu, S. (2016) 'E-CARP: an energy efficient routing protocol for UWSNs in the internet of underwater things', *IEEE Sensors Journal*, June, Vol. 16, No. 11, pp.4072–4082.

ZigBee (2008) *ZigBee Specifications*, ZigBee Alliance.