Turkish Journal of Physiotherapy and Rehabilitation; 32(3)
ISSN 2651-4451 | e-ISSN 2651-446X

SOAP And XMPP Based An Integrated Framework For Enhancing Communication Performance Of Iot Applications

J. Auxily Jovita
Research Scholar
Division of Computer and Information Science,
Annamalai University,
Annamalainagar – 608002
Email: auxilyantony@gmail.com

Dr. G. Ramachandran
Associate Professor,
Dept. of Computer Science & Engineering,
Annamalai University
Annamalainagar – 608 002.
Email: gmrama1975@gmail.com

Dr. N. Edison Rathinam
Assistant Professor,
Dept. of Computer Science,
K.C.S. Kasi Nadar College of Arts and Science
R.K. Nagar,Chennai – 600021
Email: edisonrathinam@gmail.com

ABSTRACT

In recent times, Internet of Things (IoT) is emerging as a frontier technology in implementing multiple drastic and real time changes over the environment of Information Technology. It acts as a medium amid physical and virtual world. Due to the increased amount of IoT devices, size and speed of data is increasing very high. Communication protocols are primary players of IoT system which ensures IoT data exchange and various frameworks and protocols are utilized in different patterns of messaging. While deigning and deploying as IoT devices are restricted in term of computation and processing, selection of correct communication protocol is a highly challengeable task with the consideration of features such as reliability, scalability, lightweight, interoperability, extensibility and security. In this paper to meet those design and deployment requirements in IoT system and to enhance the performance of IoT applications, a novel framework is proposed by integrating the significant features of (SOAP) and eXtensible Messaging and Presence Protocol (XMPP). Limitations of existing protocols are analyzed and compared with the proposed framework. Experimentation results provide better performance for the IoT system than the earlier works.

Key words: SOAP, XMPP, IoT

I. INTRODUCTION

Data management of IoT devices is achieved effectively by the strong contribution of improved networking technology. Rapid amplification is observed in the size and rate of data created by the IoT devices. Various communication protocols are employed in order to make effective message exchange by IoT devices. Application layer protocols of Open System Interconnection (OSI) model define the choice of the pattern of message exchange [1]. Layered architecture of IoT system is depicted in Fig.1.

Hyper Text Transfer Protocol (HTTP), Message Queuing Telemetry Transport (MQTT), Data Distribution Service (DDS), Advanced Message Queuing Protocol (AMQP), Constrained Application Protocol (CoAP), (SOAP) and Extensible Messaging and Presence Protocol (XMPP) are some examples of communication protocols which are associated with IoT applications. Few of them are similar wile considering connectivity as feature. IoT devices mainly collect data and exchange it for the entire execution of IoT system. Further, selection of communication protocol for a device is a challengeable task and it takes more time. In addition it is important to consider the hardware configurations of IoT devices and nature of data link layer protocols associated [2]-

[4]. As IoT devices are having different bandwidth, their associated data rate also seems distinct which depends on the size and hardware configuration of the IoT devices.

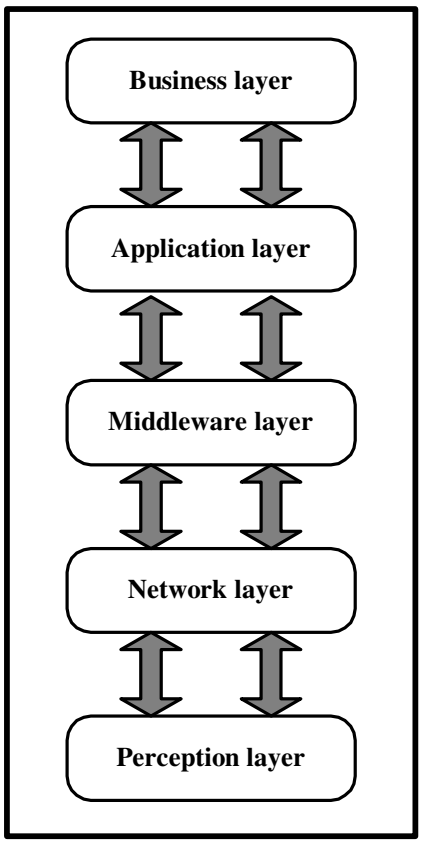

Fig.1 Layered architecture of IoT

Besides usage of application layer protocols will increase latency and affects physical data rate. To resolve this issue it is mandatory to consider physical data rate on DLL and right communication protocol during the development of IoT application. Light weighted communication protocols increase speed of data rate in IoT devices. Message Queuing Telemetry Transport (MQTT) is an example of asynchronous light weighted communication protocol which is considered as a Machine-to-Machine (M2M) or IoT connectivity protocol [5]. Performance of IoT system is primarily based on the selection of communication protocol. Deployment of appropriate communication protocol will increase reliability by decreasing the traffic of network and latency [1]. Still there is no common protocol used over various IoT environments. Selecting correct communication protocol depends on multiple aspects such as business requirements of IoT application, software and hardware configuration, standard data exchange size and others. It is essential to concentrate primary features of already existing protocols.

Many review works are carried to investigate the protocols of DLL and IoT connectivity [6]-[8]. More than understanding DLL protocols (such as WirelessHART, Sigfox, LoRa, and others), consideration of application layer protocols (such as HTTP, CoAP, MQTT, and others) is also important for the development of IoT applications [9]-[12], [13]-[19]. Thus understanding the

interaction of protocols via OSI model is necessary. Furthermore protocols of DLL and application layer have distinct objectives and the critical IoT system is considered with the requirements like Quality of Service (QoS), bandwidth, interoperability and security. Though more studies are about investigation of protocols in DLL and application layer, still they didn't define the connectivity amid protocols over OSI model. Working completely on many layers of protocol stack is needed to support the complete development of application. Those research gaps are considered and they are intended to minimize in order to increase the performance of IoT devices and IoT applications of the IoT system.

The rest of the paper is categorized as the following Sections: Section II describes the existing research works related to the communication protocols of IoT system, Section III details preliminaries (i.e., SOAP and XMPP), Section IV describes the paradigm of proposed framework and its functionalities, and Section V exhibits the performance analysis of the proposed framework. Finally, conclusion will be carried as specified in the last section.

II. RELATED WORK

Most of the research works in IoT system mainly concentrated on data collection, data analysis and integration processes. IoT data flow is systematized in a model comprised with multiple levels. Deployment of IoT reference model contains various components such as physical devices and controllers, connectivity, data collection and generalization and application. Application layer protocols have major contribution in building an effective IoT system. Features of CoAP and MQTT are analyzed and integrated for utilizing in healthcare based IoT applications [11]. Drawbacks of those two protocols are discovered and also security based case studies are observed for healthcare background. Then it concentrated to ensure security by finding the limitation in protecting patient's private data. In addition threats are categorized on the basis of the security aspects like privacy, secrecy, accessibility and integrity.

Detailed review work is carried on the existing communication protocols by Naik and their advantages and shortcomings are recognized [10]. Similarly review work of Dizdarević et al concentrated on existing protocols of application layer in the fog and cloud based systems [12]. Thangavel et al investigated both MQTT and CoAP on the basis of performance at both ends through a general middleware [20]. Utilization of both protocols is focused in wireless sensor networks (WSNs). To connect existing and current protocols middleware is proposed. Messaging in MQTT has low delay than CoAP on the basis of low packet loss rate. But in high data rate, CoAP performs better than MQTT by creating minimum traffic overhead. BeagleBoard-xM, a laptop and a notebook are used as middleware, server and Wide Area Network (WAN) emulator respectively.

Likewise the study of Tandale et al. focused on three communication protocols which are CoAP, MQTT and HTTP REST [9]. Raspberry Pi 3 is utilized for investigating the reliability and traffic of those three protocols under various network circumstances. Cellular network (4G) and high speed broadband connection are considered for study. Finally it is recognized that CoAP's performance is efficient over small payloads. A complete literature review is presented by Chen et al. [21] for evaluating the performance of the protocols CoAP, MQTT, DDS and a custom UDP-based protocol in the healthcare related application. Real time data is collected from patients to measure performance on the basis of metrics like bandwidth utilization, latency and packet loss rate. In addition performance of protocols in minimum quality wireless networks is explained in detail. Raspberry Pi model 2, Arduino Uno revision 3 and Windows laptop ASUS

Zenbook are utilized for experimental purpose. Implemented software tools are represented as Californium CoAP, HiveMQ, Mosquito and OpenDDS. Tools like TBF, NetEM and Wireshark are utilized for analyzing the performance of protocols.

Based on the results TCP based protocols such as DDS and MQTT perform better than UDP based protocols like Custom-UDP and CoAP. But considering latency UDP based protocols are better. Bandyapadhyay et al compares XMPP and CoAP protocols to determine the most suitable protocol for various application areas of IoT devices [22]. Performance evaluation is carried with the help of Android O/S and Intel X86 systems. Libcoap and Mosquito libraries are utilized by the protocols CoAP and MQTT respectively. Further famous network traffic analyzer Wireshark is also used. Comparison of protocols is done by considering the metrics such as energy utilization, bandwidth consumption and reliability. Based on the experimentation result with the consideration of optimized energy consumption, performance of CoAP is better than MQTT. Machine-to-Machine (M2M) protocols of IoT are compared their performance is analyzed [23]. Temperature data is mainly collected for examination during the use of MQTT and CoAP protocols in IoT system. It is concluded that 100% transfer of data is achieved making use of these protocols. In addition data loss rate of CoAP is minimum for small size IoT data. Network conditions decide the performance of every protocol.

While considering networking layer protocols they allow technologies to implement communication in the form of either unidirectional or bi directional [24]. Based on the size of networks network layer protocols have distinct detection techniques to locate devices. Network addressing is implemented through IPv4, IPv6 or 6LoWPAN for detection due to the different hardware characteristics like power expenditure, connectivity medium and transmission area [4]. Detecting appropriate communication medium is a challengeable task [2].Radio Frequency Identification (RFID) is an appropriate technology for detecting and following IoT devices [3]. RFID devices are generally utilized in various applications like transport, logistics, manufacture and apparatus tracking [24]-[26]. Smart devices are required for IoT system than classical detection devices. WSNs maintain high transmission area than RFID sensor networks.

Development and deployment of smart IoT applications are performed using various techniques of communication [6]. Four important techniques are classified which are device-to-device (D2D), device-to-cloud (D2C), device-to-gateway (D2G) and device-to-application (D2A). Further various features like architectural design, software representation, confidentiality, security and communication are compared. Further wired and wireless messaging protocols are also analyzed on the basis of various measures like frequency, data rate and topology of network. Through SOAP or REST services, one or multiple services are amalgamated to execute precise functionalities. IoT services should adopt interoperability. HTTP is used as standard protocol for performing RESTful services currently. SOAP services are different from RESTful services which implement communication through Simple Object Access Protocol (SOAP). Device Profile for Web Services (DPWS) which is an OASIS standard facilitates various web services functionalities like messaging, identification of resources at the ends [27]-[28]. But HTTP and SOAP are not suitable for executing services over every kind of IoT systems.

CoAP is suitable for the IoT devices with minimum power while incorporating through HTTP. As SOAP is related with big sized messages, CoAP becomes correct option for IoT devices which interoperate with HTTP in service binding. For real time service provisioning in RESTful method, CoAP is a correct choice of protocol in engaging devices with certain

constrains. XMPP is familiar in exchanging the real time data packets and also it employed with XML specification [29]-[30]. Thus XMPP seems further appropriate choice for service provisioning of IoT systems. Executing RESTful services via XMPP is not effective choice while it is not accomplished with suitable protocol design. Affording RESTful services in communication protocols will be suitable for services related to mobile devices, automation in industry, robotics and others. Moving XMPP datagrams making use of HTTP will allow the browser which are executing through JavaScript.

Other side DDS does not involve in the distribution of data through HTTP implementation [31]-[34]. This shortcoming is overcome the utilization of RESTful DDS which binds the requests from HTTP protocol. Various tool and libraries are used to support RESTful interfaces such as RoboMQ, RabbitMQ and Eclipse Paho. XMPP, CoAP, AMQP, MQTT, DDS and MQTT-SN are the different messaging protocols of IoT system which are deeply analyzed by Anusha et al. [35]. All protocols are compared with each other on the basis of performance measures like data size, packet loss rate, bandwidth utilization and latency. Further performance of every protocol is evaluated by considering the IoT application. Moreover, while considering the experimentation results XMPP performed well because messaging applications transmits XML based data through internet.

Communication protocols CoAP, MQTT, XMPP, and WebSocket are reviewed by Kayal et al. [36]. Efficient communication in constrained devices is considered as main objective. Smart parking scenario is considered as case study for comparing the protocols. While considering software sources Libcoap, Mosquito, Smack Client and HiveMQ are utilized by CoAP, MQTT, XMPP, and WebSocket respectively. CoAP provides better performance with low server consumption. Similarly XMPP also gives better result like CoAP. Environmental data is considered to analyze the IoT communication protocols [37]. Sensors are used for data collection in which environmental data like temperature, humidity and light are collected and forwarded to the server via various ALRE time protocols. Evaluation is implemented by the computed performance measures afforded by the message types of CoAP like GET, PUT and DELETE. Summary of the research works discussed in detailed in this section is described in Table.1.

III. PRELIMINARIES

Preliminaries of communication protocols SOAP and XMPP are provided in this section.

A. SOAP

SOAP (Simple Object Access Protocol) is the mostly utilized protocol which performs data exchange in structural XML message format [38]. SOAP meets the requirements of business applications effectively. It is application layer protocol which is recommended by W3C. SOAP provides message format for communication in which message converted to XML document. It also defines how message should be transferred making use of HTTP or SMTP (Simple Mail Transfer Protocol). SOAP messages are processed with set of defined rules. SOAP message is comprise with three different elements which are <Envelope>, <Header> and <Body>.<Header> defines the processing pattern of messages. <Body> element carries the primary information in SOAP specification. Structure of SOAP message is depicted in fig.2.

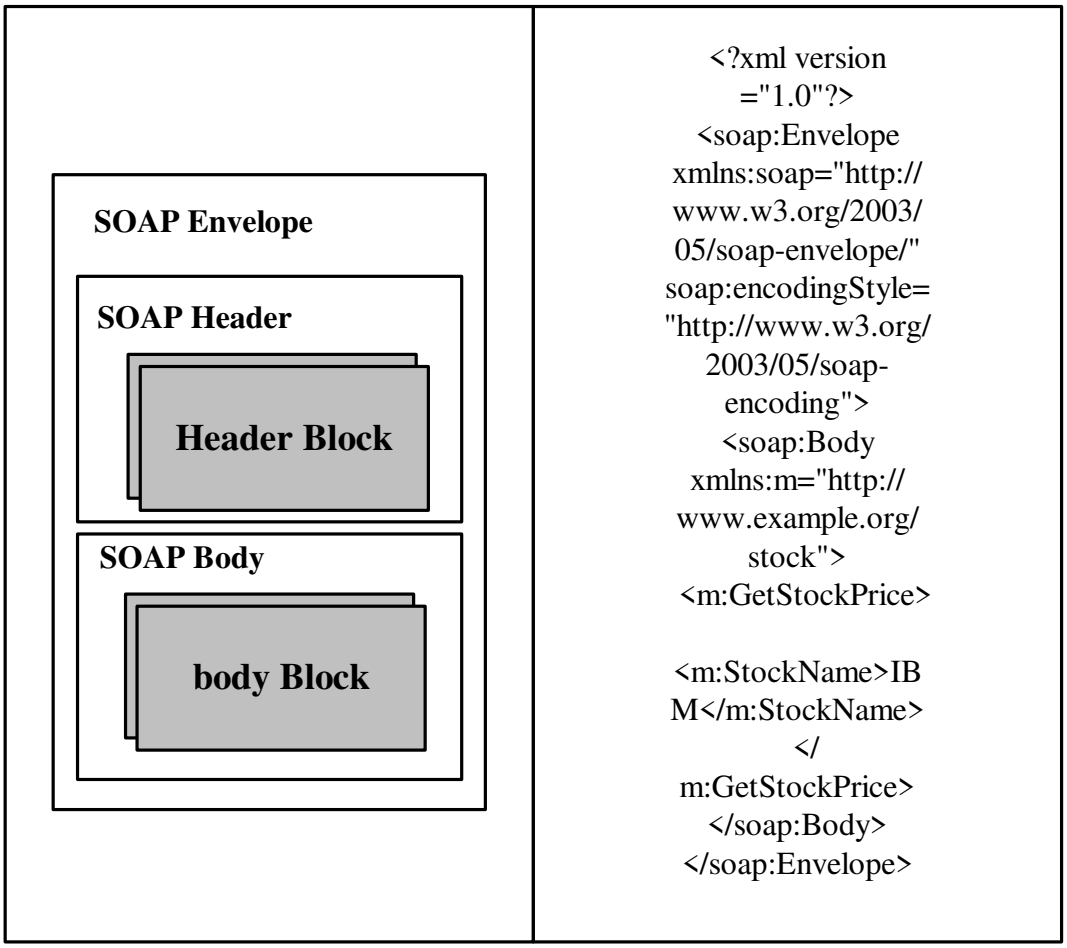

Fig.2 Structure of SOAP message and sample SOAP message

Advantages

- Ensures data exchange with high level security
- It is platform independent (i.e hardware and OS) as well as application language independent
- SOAP allows exchanging of objects amid applications with the help of XML
- It allows web service access
- It, is a simple lightweight protocol
- SOAP is a globally accepted interoperable protocol
- It supports portability and open standards utilized

Drawbacks

- Large sized messages degrade performance
- More dependency over HTTP
- Statelessness

- Serialization is implemented by value rather than reference

Papers	Protocols	H//W configuration	S/W implementation	Performance metrics	Observation
Bandyapadhyay et al. [22] (2013)	CoAP , MQTT	Android O/S, Intel X86 systems	libcoap, Mosquito	Energy utilization, bandwidth consumption, reliability	Detection of best suited protocol for constrained application areas
Thangavel et al. [20](2014)	MQTT, CoAP	a BeagleBoardxM, Laptop, Netbook	libcoap, Mosquito	Packet loss, message delay, data rate	Data transmission amid sensors are concentrated at the server's gateway
Chen et al. [21] (2016)	CoAP, MQTT, DDS, custom UDP	Raspberry Pi model 2, Arduino Uno revision 3, Windows laptop ASUS Zenbook	Californium, HiveMQ, Mosquito, OpenDDS	Bandwidth utilization, latency and packet loss	Explains functionality of protocol in low quality wireless networks
Thota et al [23] (2016)	M2M protocols (MQTT and CoAP)	-	-	data loss rate, Temperature data	Performance evaluation of protocols in different network conditions
Anusha et al. [35] (2017)	XMPP, CoAP, AMQP, MQTT, DDS, MQTT-SN	-	-	packet loss rate, message size, bandwidth utilization, latency	Functionality of every data protocol is compared
Kayal et al. [36] (2017)	CoAP, MQTT, XMPP, WebSocket	-	libcoap, Mosquito, HiveMQ, Openfire Paho Python	Response time	Performance evaluation of protocols in constrained devices
Tandale et al [9] (2017)	CoAP, MQTT and HTTP REST	Raspberry Pi 3	Cellular network (4G) high speed broadband connection	reliability and traffic	Performance evaluation of protocols in different network circumstances
Burakl et al [37](2018)	CoAP, MQTT, XMPP,	-	-	Environmental data like temperature, humidity and light	Performance evaluation of CoAP

Imane et al [11] (2018)	CoAP and MQTT	-	-	privacy, secrecy, accessibility and integrity	Protection of private data of patients
Waher et al [29] (2020)	XMPP	-	RESTful services	XMPP datagrams	XMPP performance evaluation

Table 1. Summary of recent research works

B. XMPP

Extensible Messaging and Presence Protocol (XMPP) is an open standard protocol designed by IETF which plays an important role in constructing real time applications .Various approaches of communication are used by XMPP such as instant messaging, group chat, audio and video calls, collaboration, lightweight middleware and XML data routing [39]-[41]. XMPP is an entity of XMPP Core Services and XMPP Extension Protocols (XEPs). While considering XMPP traffic, few instances of XEPs are comprised by Bidirectional-streams Over Synchronous HTTP (BOSH). XMPP communication and file transfer are carried on the basis of TCP and real time communication is supported by XML stanzas. XMPP servers permits clients for accessing server and message exchange is implemented making use of XML stanzas or tags. Furthermore XMPP allocates the existence of clients by specifying online, offline or busy. Client informs server if it is appropriate for message exchange. Fig.3 portrays the architecture of XMPP's messaging. As XMPP is a text based protocol, it performs implementations like publish to subscribe as well as client to server communication. <presence/>, <message/> and<iq/> are three distinct incorporated tags. Title and body are defined by <message/> and information or queries of sender and receiver are defined using <iq/>. XMPP is the most familiar protocol in IoT system which is implemented with similar functionalities like GET and POST methods of HTTP protocol.

Advantages

- XMPP is extendable and adjustable
- Continuous communication is implemented by several servers
- XMPP supports 3 different patterns of communication which are Client to Client, Client to Server and Server to Server
- Presence indicator presents multiple choices for messaging
- Reliable communication is offered because of the utilization of XML stanzas based on TCP protocol
- To ensure integrity and confidentiality XMPP follows the encryption procedures of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)
- Naming is implemented using XMPP jid and pubsub nodes

Drawbacks

- Server has certain limitations during communication
- Authorization to the requests of clients from servers consumes more time

- There will be delays during communication while utilizing XML Stanzas

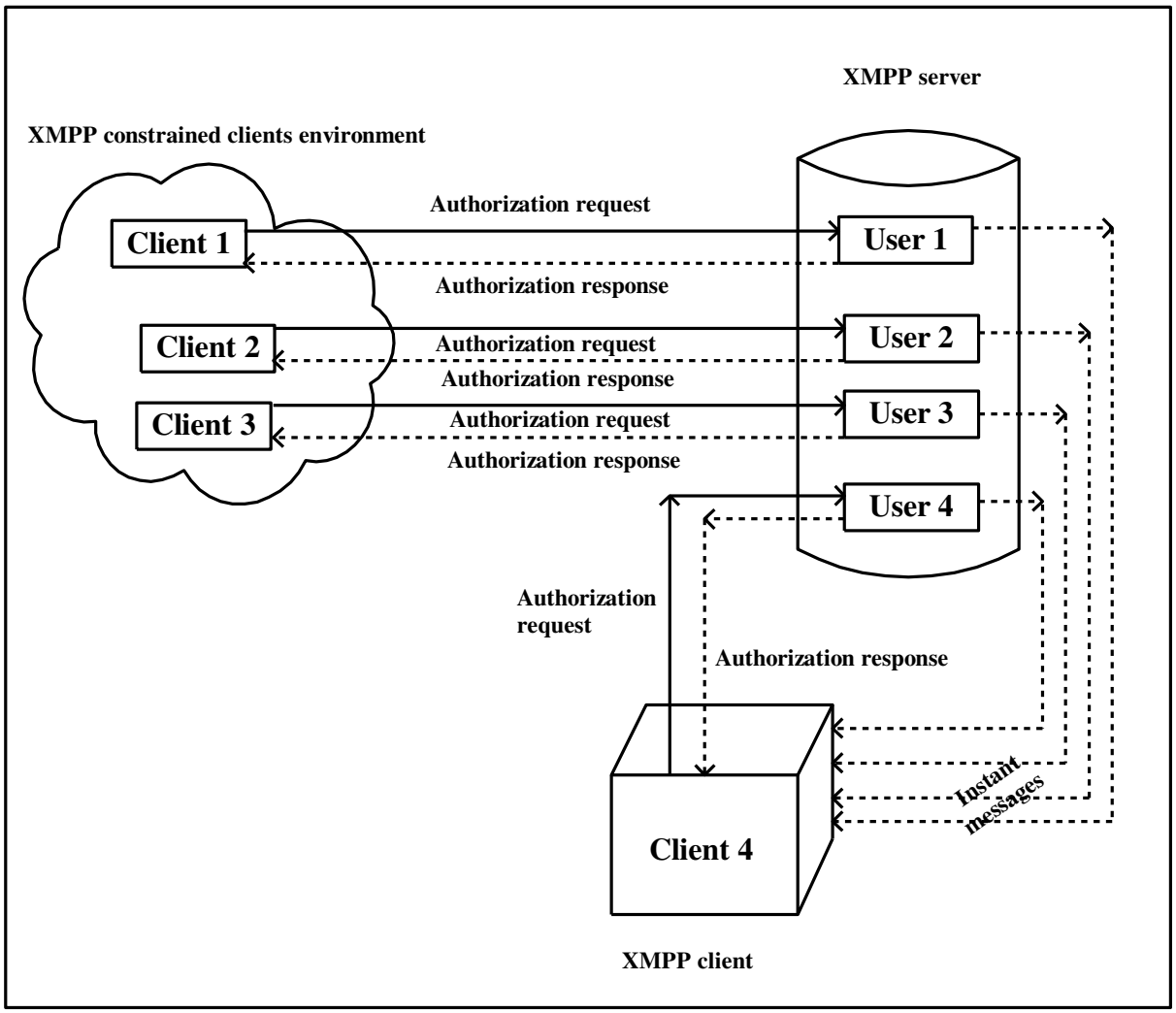

Fig.3 XMPP architecture

IV. PROPOSED FRAMEWORK

Internet of Things (IoT) system is comprised with IoT devices with different software and hardware implementations. Further IoT devices are naturally resource constrained with certain limitation regarding power, networking and processing and bandwidth. Wired or wireless technologies are employed in order to exchange data between the ends. Communication protocols contribute primarily in the data transmission of IoT systems to maintain quality efficiently. Optimization of communication protocols is required in order to accomplish the primary requirements (like processing, memory and bandwidth) of IoT devices to reduce resource consumption. Multiple existing communication protocols are developed for various purposes with different features. Identification of suitable protocol for the respective IoT system is a challengeable task. In addition environment of IoT system is advanced in heterogeneous manner, usage of more than one protocol becomes familiar. Further it again makes difficulty in the process of identifying the protocol stack by considering the environment and it will increase

the overhead of IoT devices. This issue can be overcome by selecting the suitable protocols with the consideration of two important features which are extensibility and device support.

On the basis of the detailed study of E. Al-Masri et al. characteristics for selecting IoT communication protocols includes interoperability, service provisioning, scalability, performance, reliability, security, extensibility, adaptability, incorporated application language, support to devices and protocol adoption rates [42]. In this paper earlier existing IoT communication protocols like HTTP, CoAP, MQTT, AMQP, DDS, SAOP and XMPP and their features are examined for proposing this integrated framework. Salient features of SOAP and XMPP are utilized for ensuring secure uninterrupted communication at both ends.

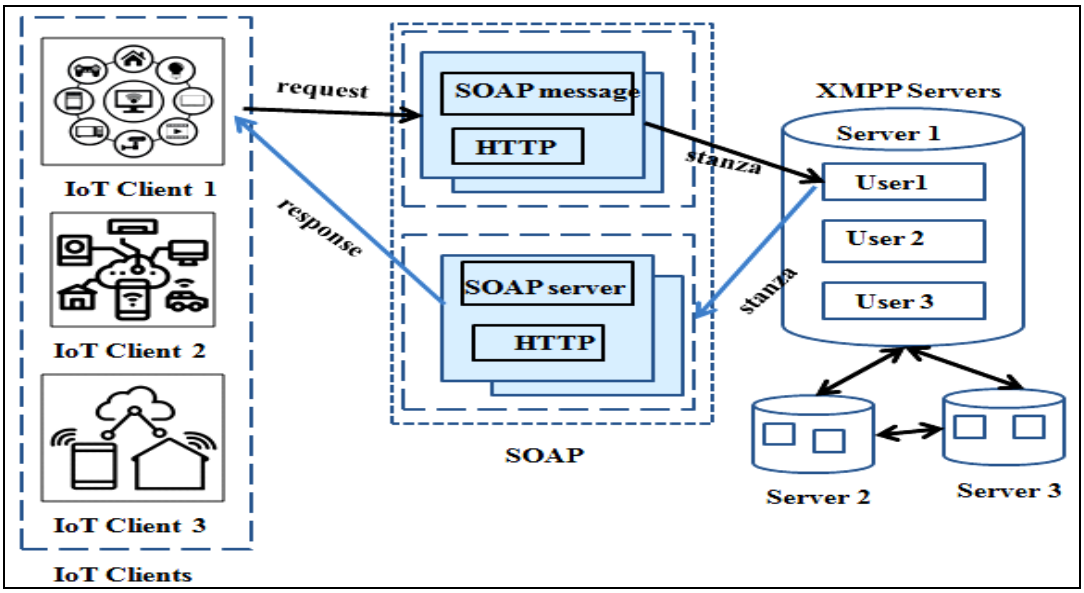

Fig.4 Proposed framework for IoT system

This proposed framework utilizes XML for message exchange which carried with the support of TCP/IP. Core services and extensible services of XMPP are implemented along with SOAP. Text based communication is pursued with the utilization of XML and end to end encryption is provided making use of two cryptographic protocols TLS and DTLS which ensures secure data communication. It will prevent from the vulnerabilities during data communication while using communication protocols SOAP and XMPP. Thus the private and confidential information of the IoT devices are protected from the malicious attackers and intruders. Possible vulnerable resources such as devices, connection medium and storage are also examined. Compromised IoT device will affect integrity where there is possible to alter or change data by the unauthorized persons. Integrity of data is assured while suing this proposed framework where authorization of requests and responses is implemented during the communication. Clients of IoT system are intended to establish secure link through this integrated framework. Requests from clients are allowed after authorizing them and responses from servers also authorized. In addition authorization tokens are issued for client application in the IoT system. It allocates password for each and every clients and for each connection establishment it should be used by

the client. Those tokens are provided in XML format. While client commences a connection to XMPP server authentication protocols will support.

Reliable transformation is enabled with the help of TCP streaming. Each and every stanzas are transferred safer through the data communication medium. Data communication follows publish/subscribe theory. Exchanging information is in encoded in the <body> element of SOAP. Occurrence of error messages is notified by fault messages. Thus SOAP messages any one type of data i.e., application specific or a fault. There is no possible for provided with both types. Usually SOAP is implemented with two different patterns of communication which are remote procedure call (RPC) and documents. In this work document style is utilized for messaging. SOAP <body> has the support of XML structure. It is mainly utilized for ensuring integrity of the original massages. Typically SOAP message is combined with HTTP. It is possible for SOAP for utilizing both the methods GET and POST. While employing GET method, responses are only considered as SOAP messages. During POST method's implementation, both request and response are considered as SOAP message. HTTP's errors and status codes are used by SOAP. In this research work also HTTP is combined with SOAP for data communication. Thus the proposed framework provides secure data transmission between the ends. SOAP ensures the speed of data transmission in the dynamic environment. Proposed integrated framework is depicted in Fig.4.

V. RESULTS AND DISCUSSION

The proposed framework is experimentally validated in the real time environment. For validation the application program is subjected to use and switch between the protocols SOAP, XMPP and Hybrid protocol which combine SOAP and XMPP. Table 2 shows the results and their performance metrics used. Fig.5 shows the schema for the performance metrics mentioned in the Table 2.

Table 2. Performance metrics

Application	SOAP	XMPP	SOAP + XMPP
Total time in ms	58.26	57.46	52.12
Instances	2	2	2
Total time (%)	35%	35%	35%
Mount (ms)	58.26	57.46	52.12
Update in ms	48.55	48.55	46.55
Render in ms of 5MB standard file	66.25	66.12	66.12
Unmount (ms)	48.55	48.55	46.55
Reloading (ms)	48.55	48.55	46.55

Fig.6 shows the overview results obtained in the cloud instance by analyzing the mentioned metrics (Refer Fig.5 for performance metrics schema).

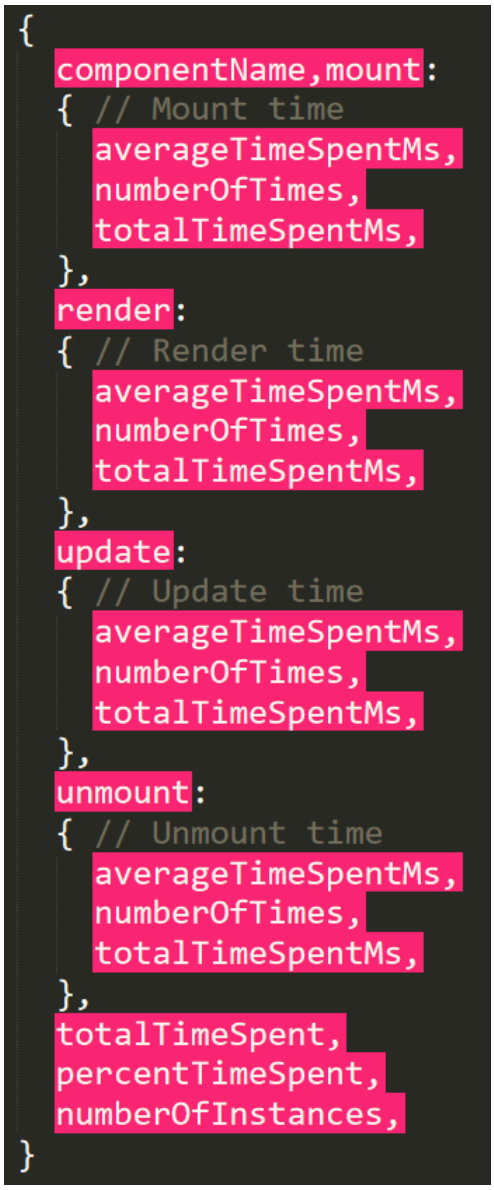

Fig.5 Performance metric - Schema

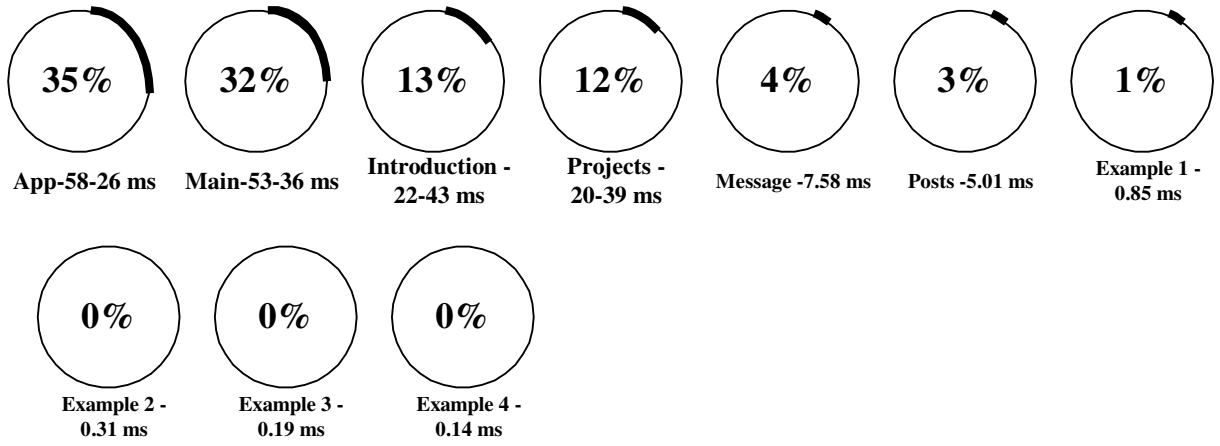

Fig.6 Screenshot of the application (SOAP + XMPP)

From the Fig.6 it is clear that the average time taken for loading the App in cloud instance is 58.26ms which is an efficient loading time with improved results of 35%. Likewise for posting a sample transaction between the IoT devices to cloud takes a least time of 7.58ms in an average internet connection operating in less than 1Mbps. The validation is carried out for four different file formats and each format succeeded with a least time for transactions. It is clearly shown in Example 1 to Example 4. Table 3 exhibits the results obtained by analyzing the mentioned metrics in Table 2.

Table 3. Results (SOAP + XMPP, SOAP, XMPP)

Feature	SOAP	XMPP	SOAP + XMPP
Total time taken by all the metrics (ms)	168.79	172.01	161
Committing changes and its reflection in ms	25.2	25.1	25.2
Committing host data and its effect in ms	58.26	57.46	52.12
Calling 1 lifecycle method (related to mounting and unmounting)	58.26	57.46	52.12
Rendering a standard data (ms)	60	60	60
Total time taken (ms)	172.07	172.07	162.02

VI. CONCLUSION

Existing communication protocols associated with IoT system are examined in detail on the basis of their performance for effective data communication. Limitations of various protocols like CoAP, SOAP, MQTT, XMPP, and DDS are analyzed mainly to provide solutions. Selecting an appropriate data communication protocol is a challenging task. Making use of more than one protocol seems familiar because of the heterogeneous environment of IoT system. Further selecting combination of protocols needs more detailed study of the system. In this paper prominent features of SOAP and XMPP are utilized to improve the performance of IoT devices in the IoT system. Secure message transformation is ensured by using the cryptographic protocols TLS as well as DTLS. SOAP helps quick uninterrupted communication. This proposed

framework will protect the private data of IoT devices of inherited from IoT system. All the data transformations are based on structured XML format amid IoT clients and XMPP server through SOAP. Proposed integrated framework's performance evaluation is carried with metrics like latency, service provisioning, security, extensibility, integrity, accessibility, resource consumption and scalability.

REFER ENCES

[1] International Organization for Standardization (ISO), Open Systems Inter- connection (OSI) Standard (35.100). Accessed: Apr. 18, 2020. [Online]. Available: https://www.iso.org/ics/35.100/x

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M.Ayyash, ``Internet of Things: A survey on enabling technologies, protocols, and applications," IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2347_2376, 4th Quart., 2015.

[3] A. Souri, A. Hussien, M. Hoseyninezhad, and M. Norouzi, ``A systematic review of IoT communication strategies for an efficient smart environment," Trans. Emerg. Telecommun. Technol., pp. 2161_3915, Aug. 2019.

[4] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, ``Internet of Things (IoT) communication protocols: Review," in Proc. 8th Int. Conf. Inf. Technol. (ICIT), May 2017, pp. 685_690.

[5] MQTT Protocol. Accessed: Apr. 18, 2020. [Online]. Available: http://mqtt.org 2020

[6] D. Mocrii, Y. Chen, and P. Musilek, ``IoT-based smart homes: A review of system architecture, software, communications, privacy and security," Internet Things, vols. 1_2, pp. 81_98, Sep. 2018.

[7] A. Juels, ``RFID security and privacy: A research survey," IEEE J. Sel. Areas Commun., vol. 24, no. 2, pp. 381_394, Feb. 2006.

[8] P. Fraga-Lamas, T. Fernández-Caramés, M. Suárez-Albela, L. Castedo, and M. González-López, ``A review on Internet of Things for defense and public safety," Sensors, vol. 16, no. 10, p. 1644, 2016.

[9] U. Tandale, B. Momin, and D. P. Seetharam, ``An empirical study of application layer protocols for IoT," in Proc. Int. Conf. Energy, Commun., Data Analytics Soft Comput. (ICECDS), Aug. 2017, pp. 2447_2451.

[10] N. Naik, ``Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," in Proc. IEEE Int. Syst. Eng. Symp. (ISSE), Vienna, Austria, Oct. 2017, pp. 1_7

[11] S. Imane, M. Tomader, and H. Nabil, ``Comparison between CoAP and MQTT in smart healthcare and some threats," in Proc. Int. Symp. Adv. Electr. Commun. Technol. (ISAECT), Nov. 2018, pp. 1_4.

[12] J. Dizdarević, F. Carpio, A. Jukan, and X. Masip-Bruin, ``A survey of communication protocols for Internet of Things and related challenges of fog and cloud computing integration," ACM Comput. Surv., vol. 51, no. 6, p. 116, 2019.

[13] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, ``Future Internet: The Internet of Things architecture, possible applications and key challenges," in Proc. 10th Int. Conf. Frontiers Inf. Technol., Dec. 2012, pp. 257_260.

[14] N. Komninos, E. Philippou, and A. Pitsillides, ``Survey in smart grid and smart home security: Issues, challenges and countermeasures," IEEE Commun. Surveys Tuts., vol. 16, no. 4, pp. 1933_1954, 4th Quart., 2014.

[15] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. Mccann, and K. Leung, ``A survey on the ietf protocol suite for the Internet of Things: Standards, challenges, and opportunities," IEEE Wireless Commun., vol. 20, no. 6, pp. 91_98, Dec. 2013.

[16] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, ``Internet of Things (IoT): A vision, architectural elements, and future directions," Future Gener. Comput. Syst., vol. 29, no. 7, pp. 1645_1660, Sep. 2013.

[17] A. Seferagi¢, J. Famaey, E. De Poorter, and J. Hoebeke, ``Survey on wireless technology trade-offs for the industrial Internet of Things," Sensors, vol. 20, no. 2, p. 488, 2020.

[18] A. Varghese and D. Tandur, ``Wireless requirements and challenges in industry 4.0," in Proc. Int. Conf. Contemp. Comput. Informat. (IC3I),Nov. 2014, pp. 634_638.

[19] Waviot. Smart Gas Metering. Accessed: Apr. 18, 2020. [Online]. Available: https://waviot. com/iot/solutions/smart-metering/smart-gasmetering, Accessed on: Apr. 18, 2020

[20] D. Thangavel, X. Ma, A. Valera, H.-X. Tan, and C. K.-Y. Tan, ``Performance evaluation of MQTT and CoAP via a common middleware," in Proc. IEEE 9th Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process. (ISSNIP), Apr. 2014, pp. 1_6.

[21] Y. Chen and T. Kunz, “Performance Evaluation of IoT Protocols under a Constrained Wireless Access Network,” IEEE 2016 International Conference on Selected Topics in Mobile & Wireless Networking (MoWNeT), Apr. 2016, pp. 1-7.

[22] S. Bandyopadhyay and A. Bhattacharyya, “Lightweight Internet Protocols for Web Enablement of Sensors using Constrained Gateway Devices,” IEEE 2013 International Conference on Computing, Networking and Communications (ICNC), pp. 334-340, Jan. 2013.

[23] P. Thota and Y. Kim, ``Implementation and comparison of M2M protocols for Internet of Things," in Proc. 4th Intl Conf. Appl. Comput. Inf. Technol./3rd Int. Conf. Comput. Sci./Intell. Appl. Inform./1st Int. Conf. Big Data, Cloud Comput., Data Sci. Eng. (ACIT-CSII-BCD), Dec. 2016, pp. 43_48.

[24] Q. Hassan, Internet of Things A to Z: Technologies and Applications. Hoboken, NJ, USA: Wiley, 2018.

[25] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, Transmission of IPv6 packets over IEEE 802.15.4 Networks, Internet Proposed Standard RFC document 4944, 2007.

[26] N. Kushalnagar, G. Montenegro, and C. Schumacher, IPv6 Over Low- PowerWireless Personal Area Networks (6LoWPANs): Overview, Assump- tions, Problem Statement, and Goals, document RF 4919, vol. 10, 2007.

[27] XEP-0072: SOAP Over XMPP. Accessed: Apr. 18, 2020. [Online]. Available: https://xmpp. org/extensions/xep-0072.html

[28] Devices Pro_le for Web Services (DPWS). Accessed: Apr. 18, 2020. [Online]. Available: http://docs.oasis-open.org/ws-dd/ns/dpws/2009/01

[29] P. Waher. XEP-0332. Accessed: Apr. 18, 2020. [Online]. Available: https://xmpp.org/extensions/xep-0332.html

[30] P. Saint-Andre, K. Smith, and R. Tronçon, XMPP: The De_nitive Guide: Building Real-Time Applications with Jabber Technologies.Newton, MA, USA: O'Reilly Media, 2009.

[31] RESTful DDS. Accessed: Apr. 18, 2020. [Online]. Available: https://code.google.com/arc hive/p/restful-dds/

[32] Eclipse Paho. Accessed: Apr. 18, 2020. [Online]. Available: https://www.eclipse.org/paho

[33] RabitMQ Multi-Protocol Support. Accessed: Apr. 18, 2020. [Online]. Available: https:// cloudamqp.com/docs/protocols.html

[34] RoboMQ over REST. Accessed: Apr. 18, 2020. [Online]. Available: https://robomq.readthe docs.io/en/latest/REST

[35] M. Anusha, E. S. Babu, L. S. M. Reddy, A. V. Krishna and B. Bhagyasree, “Performance Analysis of Data Protocols of Internet of Things: Qualitative Review,” International Journal of Pure and Applied Mathematics, vol. 115, no. 6, pp. 37-47, 2017.

[36] P. Kayal and H. Perros, “A Comparison of IoT application layer protocols through a smart parking implementation,” IEEE 2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN), Mar. 2017, pp. 331-336.

[37] Çorak, Burak & Okay, Feyza & Güzel, Metehan & Murt, Sahin & Ozdemir, Suat. (2018). Comparative Analysis of IoT Communication Protocols. 1-6. 10.1109/ISNCC.2018.8530963.

[38] https://www.w3schools.com/xml/xml_soap.asp

[39] Extensible Messaging and Presence Protocol (XMPP). Apr. 18, 2020. [Online]. Available: https://xmpp.org

[40] P. Saint-Andre, “Extensible Messaging and Presence Protocol (XMPP): Core,” Internet Engineering Task Force (IETF), Request for Comments: 6120, Mar. 2011.

[41] M. Kirsche and R. Klauck, “Unify to bridge gaps: Bringing XMPP into the Internet of Things,” 2012 IEEE International Conference on PervasiveComputing and Communications Workshops (PERCOM Workshops), pp. 455-458, Mar. 2012.

[42] E. Al-Masri et al., "Investigating Messaging Protocols for the Internet of Things (IoT)," in IEEE Access, vol. 8, pp. 94880-94911, 2020, doi: 10.1109/ACCESS.2020.2993363.