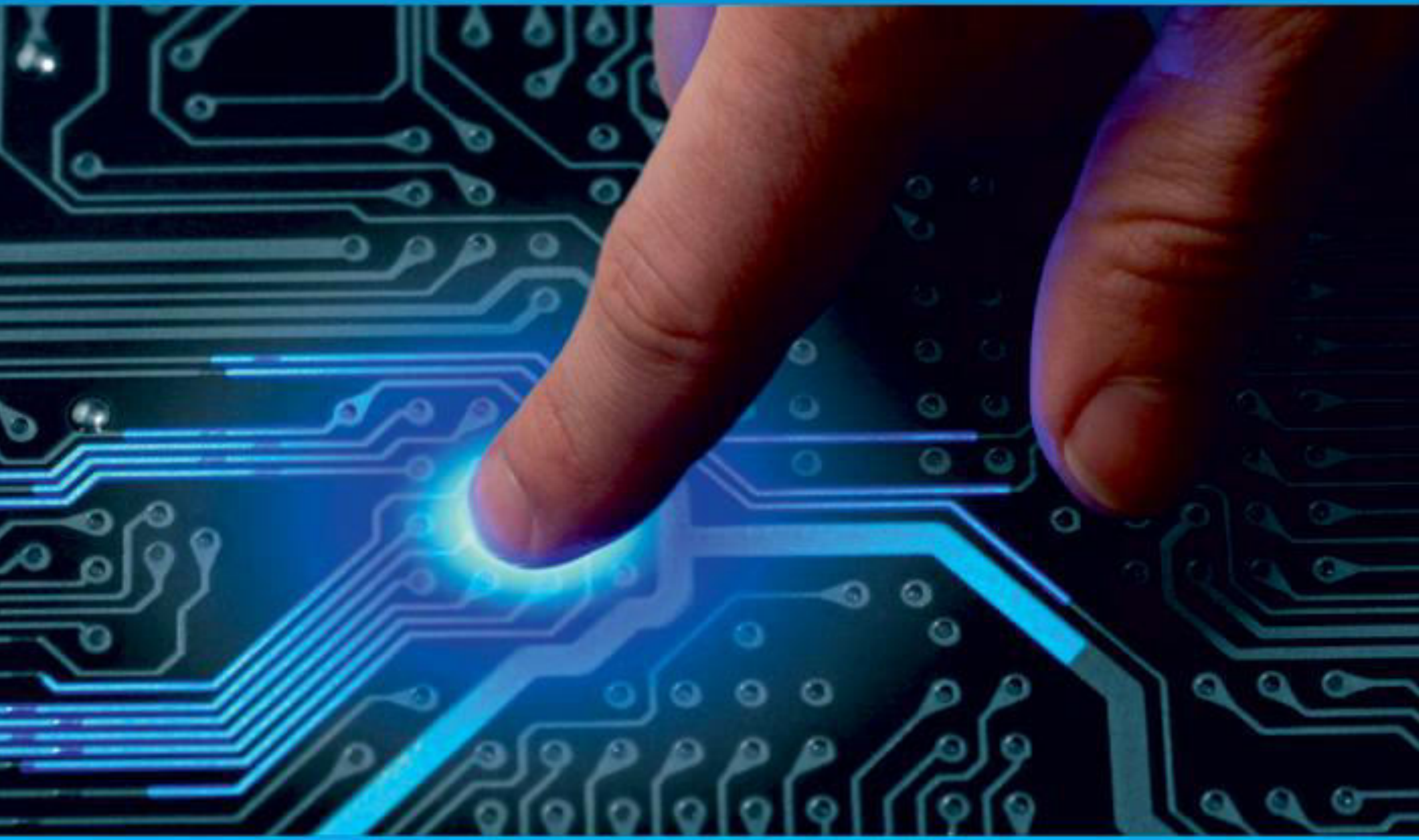
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 11, November 2021

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.542

9940 572 462 | 6381 907 438 | ijircce@gmail.com | www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 11, November 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0911033 |

Secure MQTT for Internet of Things using Dynamic S-box AES

Vaishali R Verma

Dept. of Electronics and Telecommunication, Deogiri Institute of Engineering, Aurangabad, India

ABSTRACT: The Publish-Subscribe messaging paradigm is widely used as one of the communication models for Internet of Things (IoT).Here, Users are more interested in sharing and retrieving information and careless about which specific endpoint is holding the information. Of various publish-subscribe protocol promoted, MQTT is extremely lightweight and widely used Publish Subscribe connectivity protocol for Internet of Things. The major drawback of MQTT protocol is that it has limited security features. Hence, these protocols need to address security issues for IoT. In this paper, we present a solution to provide access control and confidentiality of the information exchanged in an MQTT based IoT system. Our approach provides a solution of applying Attribute-based Encryption(ABE) and Dynamic S-Box Advanced Encryption Standard(AES) for payload encryption in MQTT. Further, we evaluated our proposed scheme through simulation.

KEYWORDS: ABE; AES; Cryptography; Dynamic S-Box; IoT; MQTT; Security

I. INTRODUCTION

Rapid growth of Internet of things and data/content-centric applications has motivated researchers to amend and modernize the way information is stored and delivered on the Internet [1]. Users are more interested in sharing and retrieving information and careless about which specific end point is holding the information. At the application level, users are more interested in the content they are interested in retrieving and less about where that content can be found [1]. For Internet of Things, the Publish-Subscribe messaging paradigm is widely used as one of the content-centric communication models which provide features such as loose coupling between clients and server, dynamic and flexible information exchange, and reusability. Of various publish- subscribe protocol promoted, MQTT is extremely lightweight and widely used Publish- Subscribe connectivity protocol for machine-to-machine (M2M)/"Internet of Things". The main drawback of MQTT is that it has limited security and note that it's user's responsibility to address this security issue. In this paper, we are augmenting security to MQTT by using MQTT payload encryption. In this application, specific data is encrypted on the application level. The main advantage of payload encryption is it provides end-to-end message security and in situations where Transport Layer Security (TLS) cannot be used [2]. Attribute Based Encryption (ABE) [3] is classic encryption algorithm used for publish-subscribe architecture [4]. The main advantage of ABE is that it supports broadcast encryption.

Hence with one encryption message is delivered to multiple intended users. In this paper, we present a new security solution to MQTT by using ABE in combination with Dynamic S-Box AES. The contribution of this paper is (i) to enable security feature for MQTT, (ii) to use hybrid or composite security for MQTT payload encryption and (iii) to study the feasibility of our proposed methodology through simulation. We implemented and evaluated our proposed solution in Java and performance evaluation parameter includes execution time, CPU and memory usage. Remaining paper is structured as follows: Section II provides a brief introduction of MQTT protocol and Attribute Based Encryption followed by related work in section III. We describe the concept and system design in Section IV. Section V gives implementation and results. Section VI gives a brief summary and conclusion.

II. BACKGROUND

This section provides background information on the protocol and techniques used in our solution.

A. MQ Telemetry Transport (MQTT)

MQTT is an extremely lightweight publish subscribe "Internet of Things" connectivity protocol. The publish-subscribe model is an asynchronous communication paradigm where publishers (senders) and subscribers (receivers), exchange messages without establishing direct contact. Publishers do not send messages directly to subscribers, instead, intermediate broker is responsible for delivering the messages to the intended subscribers. In order to receive messages, subscribers need to register to a broker through a subscription known as a topic. In

MQTT, Topic, a UTF-8 string, is used to publish and to subscribe. Topic strings are is used by the broker to filter messages for each connected client. Fig. 1 shows a simple

publish-subscribe network that forwards messages from publishers to interested subscribers. The publisher publishes the message under a particular topic. Broker use topic string to filter messages and publish it to corresponding subscribers.

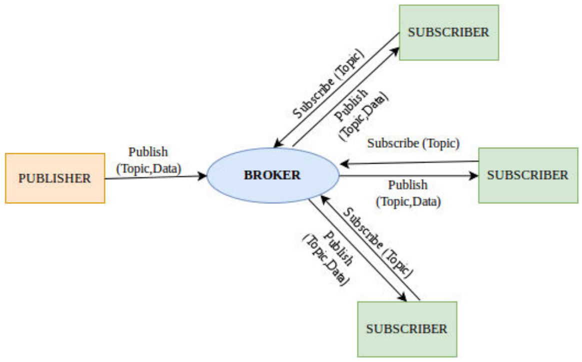


Fig. 1. The Publish/Subscribe Model.

As a transport protocol, MQTT relies on TCP, and by default the connection does not use encrypted communication. It is the implementers' responsibility to provide appropriate security features.

B. Attribute Based Encryption Attribute-based encryption (ABE) is asymmetric or publickey encryption for data/content-centric security. Senders (publishers) and receivers (subscribers) do not need to share secret keys, thus simplifying key management for large scale dynamic applications. In ABE a set of attributes (example name or designation of person) or policies defined over attributes is used to encrypt messages. A receiver can decrypt a ciphertext with a secret key define over valid access policy. Key Policy Attribute Based Encryption (KP-ABE) and Ciphertext Policy Attribute Based Encryption (CP-ABE) are two most popular ABE scheme. In KP-ABE [6], the plaintext is encrypted using a set of attributes and a user's secret key defines access policy. In CP-ABE [7], a user secret key is defined based on a set of attributes and a ciphertext specifies an access policy. The access policy or structure is usually represented as a tree, allows expressing any monotone access formula consisting of AND, OR, or threshold gates. An important security feature of ABE is collusion-resistance. A user that holds multiple keys should only be able to access data if at least one individual key grants access. The attributes used in ABE can be in the form single string (e.g. "attr_str","name", "24") or an attribute string with values (e.g. "attr_str = attr_value", "age=18"). In our implementation we have used both variations of attributes for encrypting MQTT payload. And for worst case analysis for we have used "AND" logic for access policy.

III. RELATED WORK

The concept of publish-subscribe paradigm was proposed in [4].The publish-subscribe paradigm allows loose coupling between communicating entities [5] thus providing dynamic and flexible information exchange between a large number of entities. It allows asynchronous communication between devices which do not need to be online at the same time to communicate. Security of information in the publish-subscribe system is important because once the data is published on the network, publishers lose all control over who gets access to their data. Many techniques are used to provide confidentiality and access control of information exchanged. Attribute Based Encryption (ABE) is one such encryption that provides both confidentiality and access control of data/information.

The very first concept of ABE was proposed by Sahai and Waters [3], in which client or user is identified by a set of attributes. A user encrypts the plain text with sets of attributes. A user can decrypt a particular ciphertext, only if their attributes match. Key Policy Attribute-Based Encryption (KPABE) [6] and Ciphertext Policy Attribute-Based Encryption

(CP-ABE) [7] are two main types of ABE. In KP-ABE, the plaintext is encrypted using a set of attributes and users secret key defines access policy. CP-ABE is dual of KP-ABE in the sense that users secret key is defined based on a set of attributes and a ciphertext specifies an access policy. In [1][15], M. Ion provides novel scheme that supports confidentiality of messages using ABE. In this messages are encrypted using AES and AES key is then encrypted using ABE. Author has used both KP-ABE and CP-ABE for implementation. Authors of [8] has implemented the scheme provided in [1] and evaluated the scheme under different performance parameter like execution time, CPU and memory usage under different security level. In [9], Tariq proposed a distributed algorithm to communicate events in a publish subscribe system in term of delay and available bandwidth. In [10], author introduced a new signature scheme called ABSIGN for KP-ABE, which enables the verifier to ensure that a signature is produced by a user whose access policy is satisfiable by a set of attributes without learning the signer's identity. [11] gives detailed performance evaluation of ABE on both smartphone and Internet of Things devices. In this paper, we are enabling security in MQTT (publish subscribe) protocol. [8] and [11] shows that KP-ABE outperforms CP-ABE. Hence to enable security in MQTT we have used KP-ABE. To make it lightweight we have used KPABE [6] over lightweight ECC [12]. To further increase the security strength we have used Dynamic S-Box AES ([13] shows that performance of Static S-Box AES and Dynamic SBox AES are relatively same with increase security).

IV. METHODOLOGY

Here, we assume an honest-but-curious threat model for publishers, brokers, and subscribers which means that they follow the protocol correctly, but are curious to learn as much as possible about the exchanged messages. We assume there is external Trusted Authority which generates encryption and decryption keys used to protect data from unauthorized access. The authority does not misbehave and is trusted by all the entities of the system. To provide payload encryption in MQTT, we have used KP-ABE for our solution. Note, we have to skip CP-ABE in proposed flow as in [8][11], they have shown KP-ABE outperform CP-ABE and to reduce lengthy- ness and focus more on advanced level encryption for IOT. For broadcast encryption, the authors of [6] suggest use a symmetric key for the message to be broadcast and then the same symmetric key is encrypted using KP-ABE with data attribute associated with the broadcast message. In our system we have used dynamic S-Box AES as a symmetric key encryption to encrypt the MQTT payload and AES with dynamic S- Box key is encrypted using KP-ABE scheme. Thus the system provides confidentiality of message using dynamic S-Box AES and fine-grained access control using KP- ABE. Note that we have used dynamic S-Box AES instead of static S-Box as [13] shows the performance of both is same but provide a higher level of security. Fig. 2 shows the flow of our proposed system. Our proposed system includes three phases.

1) Setup Phase:

 External trusted key authority runs KP ABE setup algorithm and generates a public key and a master secret key. Key authority publish public key to all clients devices.

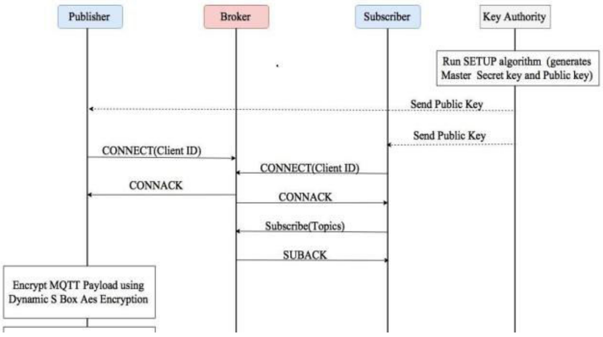
 All client devices (publishers and subscribers) sends CONNECT packet with their own unique client ID.

2) Encryption Phase

 Publisher encrypts the message to be published using Dynamic S-box AES encryption algorithm.

 Publisher then perform KP-ABE encryption
algorithm with Data Attributes, public key and Dynamic S-box AES key as input.

 Publisher then publish the encrypted key and encrypted message for a particular topic.



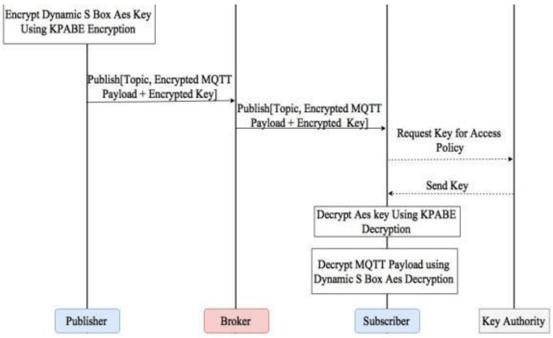

Encrypt Dynamic S Box Aes Key Using KPABE Encryption
Publish[Topic, Encrypted MQTT Payload + Encrypted Key]
Publish[Topic, Encrypted MQTT Payload + Encrypted Key]
Request Key for Access Policy
Send Key
Decrypt Aes key Using KPABE Decryption
Decrypt MQTT Payload using Dynamic S Box Aes Decryption
Publisher
Broker
Subscriber
Key Authority

Fig. 2. Flow diagram of proposed secure MQTT.

3) Decryption Phase

□ The subscriber sends a request to key authority to generate a private key for its own defined access policy.
□ For valid access policy, Subscriber then runs KPABE Decryption algorithm with received encrypted key, public key and private key as input.
□ With decrypted key, Subscriber decrypt message using Dynamic S-box AES decryption algorithm.

V. IMPLEMENTATION AND ANALYSIS

A. Experimental Setup

Our proposed scheme is implemented in Eclipse Java platform. The experimental setup consists of a publisher, broker and subscriber. We have used MQTT version 3.1.1[14]. Eclipse Paho Java Client version 1.1.0 [17] is used to implement publisher and subscriber client and open source Mosquitto version 3.1 [18] is used for broker. All publisher, subscriber and broker are present on local host. For payload encryption and decryption we have implemented AES with Dynamic S-Box as described in [13]. To make KP-ABE scheme lightweight for resource constrained devices, we used ABE scheme based on Elliptic Curve Cryptography (ECC)[12][16]. We have tested our scheme on the laptop. The laptop runs 64 bit Windows 7 operating system, Intel Core Duo CPU @2.20 GHz, 3 GB RAM. We evaluated our scheme in terms of Execution time for encryption and decryption, average CPU load and average memory usage. All this operation depends on both KP-ABE operation and dynamic S-Box AES operation.

B. Performance Analysis

1) Execution Time:

Fig. 3. represent the average execution time for Key Generation for KP-ABE operation. Fig. 4 and 5 represent the average execution time for Encryption and Decryption operations. Here encryption operation includes both encryption time of KP-ABE encryption operation and Dynamic S-Box AES encryption. Similarly, decryption operation includes both decryption time of KP-ABE decryption operation and Dynamic S-Box AES decryption. Results have been obtained as an average of five iterations for each operation and varying the number of attributes from one to ten. The result is of 128-bit security level. To compare our proposed scheme with the implementation in [8], we rely on the numbers and graphs reported in [8]. As seen from Fig. 3, 4 and 5, the time required for three operations directly depends on the number of attributes. KP- ABE KeyGeneration operation requires on an average less than 2 seconds as compared to 2.5 seconds in [8]. Dynamic S-Box encryption and decryption operation require an average time of 2 milliseconds. Total encryption operation time requires an average less than 1.5 seconds as compared to 2 seconds in [8]. Decryption operation time in [8] requires 2 seconds and total decryption operation time for our scheme requires an average less than 1 second for ten attributes. Note that the execution time for all depends on a number of attribute and length of each attribute.

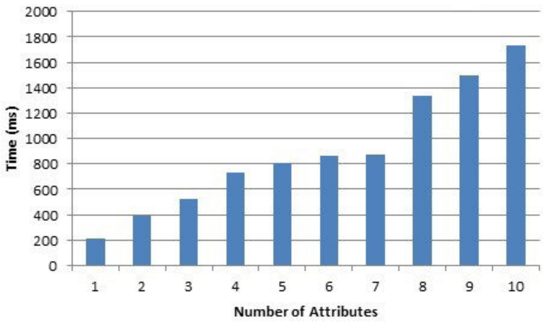


Fig. 3. Average KeyGeneration Execution Time

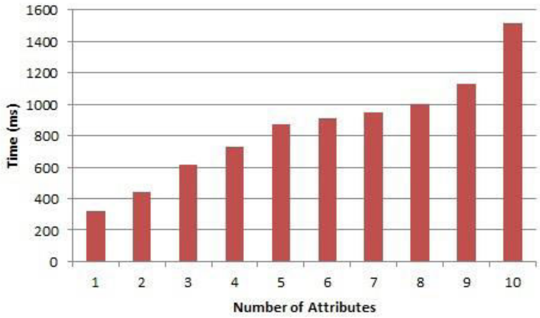


Fig. 4. Average Encryption Execution Time

From Fig. 6 and 7, the CPU utilization for encryption and decryption operation remains on an average of 50 % similar to [11]. The operations completely utilize one of two CPUs provided by the underlying platform.

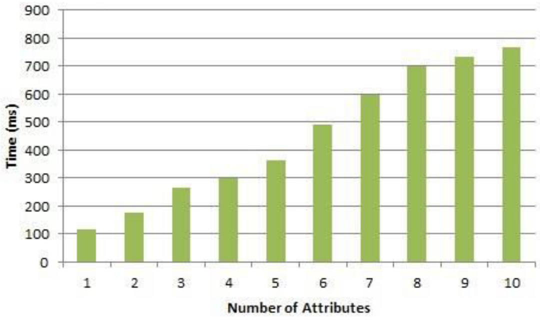


Fig. 5. Average Decryption Execution Time

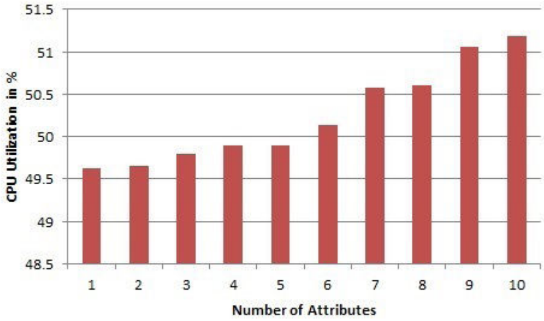


Fig. 6. Average CPU Usage for Encryption

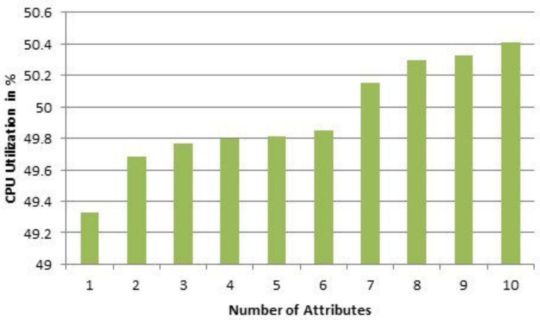


Fig. 7. Average CPU Usage for Encryption

3) Memory Usage:

Fig. 8 and 9 shows the average memory usage for encryption and decryption operation. Encryption operation uses on an average of less than 11 MB of RAM space and decryption operation uses on an average of less than 13 MB of RAM space for ten attributes similar to the number in [8].

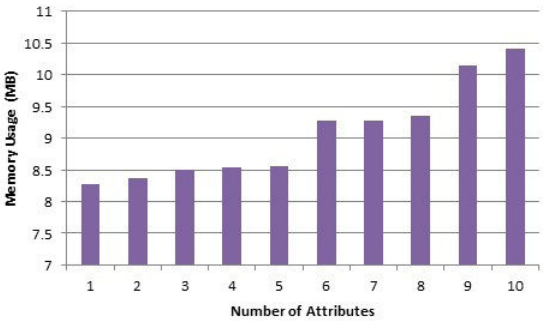


Fig. 8 Average Memory Usage for Encryption

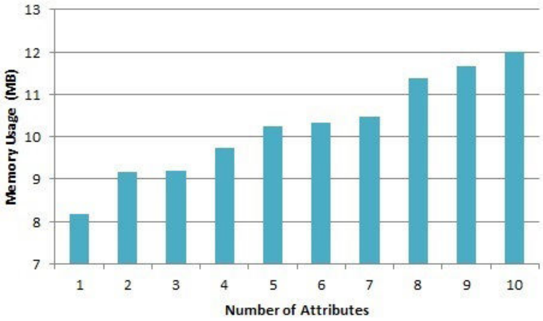


Fig. 9. Average Memory Usage for Decryption

VI. CONCLUSION

Due to rapid growth in Internet of Things and data-centric application, use of publish–subscribe paradigm is increased tremendously. MQTT, an publish-subscribe protocol, though lightweight for Internet of Things environment, has limited or devoid of security. In this paper, we enabled the security in MQTT protocol which is widely used as publish-subscribe paradigm for Internet of Things. We used hybrid or composite security for payload encryption for MQTT. Our implementation uses KPABE with lightweight ECC along with Dynamic S-box AES to further increase the security strength. KP-ABE [6] is a powerful cryptosystem that allows fine-grained access control over data and Dynamic S-Box AES provides strong confidentiality of the message. Thus, our proposed security solution provides both confidentiality and fine-grained access control of data. We did performance proposed scheme in terms of execution time, CPU and memory usage. Overall we can conclude that our proposed solution scheme provides good performance in terms of all performance parameters. Future work includes evaluation of our proposed secure MQTT protocol on IoT platform in real time.

REFERENCES

[1] M.Ion, "Security of Publish/Subscribe Systems," Ph.D. Thesis, University of Trento, 2013. http://eprints phd.biblio.unitn.it/993/.
[2] MQTT Security Essential, http://www.hivemq.com/mqtt securityfundamentals/.
[3] A. Sahai and B. Waters, "Fuzzy Identity-based Encryption," *in Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques,* ser. EUROCRYPT'05 ,Berlin, Heidelberg, , pp. 457–473, 2005.
[4] A. Carzaniga, D.S. Rosenblum and A.L.Wolf, "Design and evaluation of a wide-area event notication service," *ACM Transactions on Computer Systems (TOCS),* vol. 19, no. 3, pages 332-383, 2001.
[5] P.T. Eugster, P.A. Felber, R. Guerraoui and A.M. Kermarrec, "The many faces of publish/subscribe," *ACM Computing Surveys (CSUR),* vol. 35, no. 2, page 131, 2003.
[6] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. Of ACM CCS*, pp. 89–98, 2006.
[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute based encryption," in *Proc. of IEEE SP*, pp. 321–334, 2007.
[8] X. Wang, J. Zhang, E. Schooler, and M. Ion, "Performance evaluation of Attribute-Based Encryption: Toward data privacy in the IoT," *in Communications (ICC), 2014 IEEE International Conference on*,pp. 725–730, June 2014.
[9] M. A. Tariq, "Non-functional Requirements in Publish/Subscribe Systems," Ph.D. dissertation, University of Stuttgart, Germany, August 2013.
[10] S. Zarandioon, D. Yao, and V. Ganapathy, "K2C: Cryptographic cloud storage with lazy revocation and anonymous access," *in Proceedings of the 7th International ICST Conference on Security and Privacy in Communication Networks (SecureComm)*, pp. 491-510, 2011.
[11] M. Ambrosin, M. Conti, T. Dargahi, "On the feasibility of attributebased encryption on smartphone devices", *IoT-Sys 2015*, 2015.
[12] B. S. Adiga, P. Balamuralidhar, M. A. Rajan, R. Shastry, and V. L.Shivraj, "An Identity Based Encryption Using Elliptic Curve Cryptography for Secure M2M Communication," in Proceedings of the First International Conference on Security of Internet of Things, ser. SecurIT'12. ACM, pp. 68–74, 2012.
[13] S. Arrag, A. Hamdoun, A. B. Tragha, and S. E. Khamlich, "Implementation of Stronger AES by using Dynamic S-Box Dependent of Master key,*" Journal of Theoretical and Applied Information Technology* ,vol. 53, no. 2, Jul 2013.
[14] D. Locke, "MQ Telemetry Transport (MQTT) V3.1 Protocol Specification,"http://www.ibm.com/developerworks/library/ws-mqtt/, 2010.
[15] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the Decryption of ABE Ciphertexts," In *USENIX Security Symposium*, page 3, 2011.
[16] M. Ion, G. Russello, and B. Crispo, "Supporting Publication and Subscription Confidentiality in Pub/Sub Networks," *in Security And Privacy in Communication Networks,* ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 50, pp. 272–289, 2010.
[17] Eclipse Paho Client, https://eclipse.org/paho/.
[18] Eclipse Mosquitto Broker, https://mosquitto.org/.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

☎ 9940 572 462 6381 907 438 ✉ ijircce@gmail.com

Scan to save the contact details

www.ijircce.com