

STEM : Secure transfer of encrypted message in IoT networks

Sonam [†]

Rahul Johari ^{*}

SWINGER: Security, Wireless, IoT, Network Group of Engineering and Research

University School of Information, Communication and Technology

Guru Gobind Singh Indraprastha University

Sector 16C

Dwarka 110078

Delhi

India

Abstract

IoT stands as a dynamic domain within technology and research, facilitating organizational decision-making and aiding individuals in problem-solving by linking sensors and devices to the internet for data exchange between clients and servers. Despite numerous approaches attempted by researchers, such as authentication via login IDs and encryption through attribute-based topics or the AugPake protocol, attackers can exploit vulnerabilities at various points along the communication pathway, posing a significant threat. To mitigate this risk, a novel routing strategy called STEM (Secure Transfer of Encrypted Messages) is proposed for IoT networks. STEM encrypts transmitted messages and routes them using MQTT (Message Queue Telemetry Transport) from publishers to subscribers. Encryption is accomplished using cipher techniques, with key exchange facilitated by the Diffie-Hellman algorithm. Simulation using the ECLIPSE PAHO Java Client has demonstrated promising and effective results.

Subject Classification: 68M25 Computer security.

Keywords: Internet of Things, Publisher, Subscriber, Broker, Topics.

1. Introduction

Before delving further, it's essential to grasp the importance of the MQTT (Message Queuing Telemetry Transport) Protocol within the IoT

[†] E-mail: sonammathur57@gmail.com

^{*} E-mail: rahul@ipu.ac.in (Corresponding Author)

landscape. MQTT serves as a lightweight protocol specifically tailored for low-power devices [3]. It operates as a machine-to-machine connectivity protocol within the IoT framework. Notably, MQTT adopts an exceptionally lightweight publish-subscribe communication model, making it particularly suitable for establishing connections in remote or resource-constrained environments [4]. In the context of the current research endeavor, namely the STEM (Secure Transfer of Encrypted Messages) project, an IoT network routing strategy is introduced. This strategy entails encrypting transmitted messages and subsequently routing them through MQTT from the publisher's origin to the intended subscribers. In the pub/sub communication model, publishers and subscribers don't know each other's IP address or port. They do not have to be actively connected at the same time. Sending and receiving happen at their own pace. It guarantees the in-order transmission of a packet and also maintains the ordering if it has a larger set of Brokers and Publishers attached to it. It can support a message of 256 MB. The topics can also be used as key-value storage which means as soon as the publisher updates the topic all the subscriber gets notified automatically. MQTT has a built-in quality of service level with values ranging from 0 to 2. Figure 1 represents a diagram depicting STEM-oriented MQTT Routing facilitating Publisher-Broker-Subscriber Communication.

2. Literature Survey

In [5], the authors introduced a novel approach, combining identity-based anonymity and cryptographic authentication within IoT infrastructure. This system guarantees the security and privacy of data exchanged between IoT devices and service providers through advanced encryption techniques. Moreover, it streamlines authentication processes, enhancing system efficiency while maintaining robust security measures. By safeguarding user data and fostering reliable interactions, this proposed technique represents a valuable advancement in securing IoT environments amid their expanding presence.

In [6], the authors investigated challenges associated with 5G-IIoT routing and proposed a solution in the form of the Quality of Service and Data Privacy-aware Routing Protocol (QoSPR). To tackle these routing issues, the authors introduced QoSPR, which integrates considerations for both quality of service and data privacy. The authors also implemented modifications to the community detection algorithm, Info-map, to effectively partition the routing region into optimal subdomains. Building

upon this foundation, they employed a deep reinforcement learning algorithm to develop a gateway deployment model aimed at enhancing load balancing and reducing latency.

In [7], the authors devised a secure routing method rooted in the LEACH algorithm. Their approach involves segmenting images and messages into sub-components, representing them as matrices, and aggregating them to form a steganographic image. This steganographic image is then encrypted using an appropriate encryption algorithm. Furthermore, they integrated intelligent agents into all sensor nodes within the network to enhance the reliability of encrypted image communication among neighboring nodes.

3. Problem Statement

There is a centralized Broker that is a server in an IoT network whose job is to transfer the data with security between publisher and subscriber [8]. There is a set of publishers who publish data on the Broker over the topics that are registered by them. Publishers can have a sensor that collects data from its environment like temperature, pressure, and humidity in case of Weather Applications. There is a set of subscriber(s) who subscribe to topics which are key storage elements so as soon as the publisher updates anything in the topic they get notified immediately.

The Cipher technique has been used to encrypt the plaintext data transmitted from publisher to Broker and from Broker to subscriber. The simulation is done to show how the publisher gathers the data from its

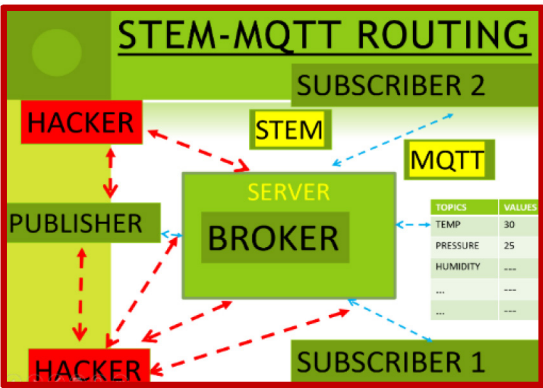


Figure 1

Diagram depicting STEM-oriented MQTT Routing facilitating Publisher- Broker-Subscriber Communication

surroundings, uses it as data for its topic, encrypts the data, and publishes it to the Broker, and the Broker sends the same encrypted data to all the subscribers who subscribe to the same topic and later they decrypt the data using the key.

4. Methodology Adopted

The implementation begins with the configuration of the Broker. In this implementation, the Broker is the mediator between publisher and subscriber which is running on a system having I7-6700HQ @2.60 (Octa Core) processor and 16GB RAM with 960GTX 4GB Graphic Card. The implementation simulates the transfer of weather-related data from publishers to subscribers through the Broker. The main aim is to transfer this data using the encryption technique [9] so for that the Diffie-Hellman Algorithm was used to transfer keys and encoded cipher technique using JSON message transfer technique in which the message is transferred in string format. The dataset utilized for simulation purposes was categorized into 10 distinct topics, including Temperature, Mean Sea Level Pressure, Total Precipitation (high resolution), Relative Humidity, Total Cloud Cover, Sunshine Duration, Shortwave Radiation, Wind Direction, Wind Speed, and Wind Gust. Each publisher publishes data for its service providing the topic of weather data, hence the number of publishers is fixed which is 10.

At the Broker, all the topics were defined. There can be any number of subscribers in the network. Subscribers register with their IDs and have 70 percent of chance for subscribing to each topic defined in the Broker. Subscribers were connected to the Broker and also for each topic subscribed by the subscriber, an entry was added to the Subscriber list at the Broker, which includes the subscriber name against the topic name. Every Subscriber and Publisher connected to the Broker exchanged the keys using the Diffie-Hellman Key Exchange Algorithm. A total of three keys were exchanged. The first key was used to find the encryption technique. The second key was used as a public key for encrypting the message. The third key was used in the case if the Cipher Technique was Affine, Affine Cipher needs two keys to work. When all key exchange was done, a table was created by each publisher and subscriber where they store these keys.

For simulation, each publisher was executed on individual threads. On that thread, a publisher publishes data every second for 360 times. So, the simulation ran for a minimum of 360 seconds or 6 minutes.

For publishing, a publisher selects a topic and asks for a list of subscribers subscribed to that topic to the Broker. The Broker then forwards the list of subscribers registered to that given topic. Then the publisher communicates with each subscriber given by the Broker. In that communication, the publisher checks if the keys were available. If no, then a set of keys was created and exchanged using the Diffie-Hellman Key Exchange Algorithm. Otherwise, the message was encrypted using keys available and encoded in a JSON (JavaScript Object Notation). The JSON object also contains the Sender Client ID, Receiver Client ID, and Message Type. This object was converted into a string object and then passed as the MQTT message on the "Message Channel". The subscriber then receives the message from the Message Channel and decrypts the messages using the cipher technique that was mentioned in its key inventory for Message Sender.

5. Algorithm

The STEM algorithm(s) 1, 2, and 3 give the step-by-step sequence of the implementation of the transfer of message and key using Broker in IoT networks. Figure 2 presents a comprehensive flow diagram with all the steps.

Algorithm 1: IoT Network with Encrypted Message

Result: Encrypted Message Has Been Delivered

X: Number of Subscribers;

Y: Number of Publishers;

CT: [C1, C2, ...Cn] Array of CipherTechnique available;

topic: Services;

n: Number of Cipher Technique

1. Set ConnectionProbability for topic Subscription
2. For $i \leftarrow 0$ to $n-1$;
3. Subscriber[i] $\leftarrow$ CreateClient(SubscriberId);
4. For each topic;
5. If(RandomProbability $\geq$ ConnectionProbability);
6. RegisterList $\leftarrow$ Broker.Register(Subscriber,topic);
7. For $i \leftarrow 0$ to $Y-1$;

8. Publisher[i] $\leftarrow$ CreateClient(PublisherId);
9. For i $\leftarrow$ 0 to Y-1;
10. For i $\leftarrow$ 0 to X-1;
11. K1 $\leftarrow$ CreateNewKey(Publisher[i],Subscriber[j]) for Cipher Technique Decision;
12. K2 $\leftarrow$ CreateNewKey(Publisher[i],Subscriber[j]) for Cipher-Key;
13. K3 $\leftarrow$ CreateNewKey(Publisher[i],Subscriber[j]) for Second Cipher-Key;
14. Put Publisher[i]Id, K1, K2, k3 in key inventory of Subscriber[j]id;
15. Put Subscriber[j]Id, K1, K2, K3 in key inventory of Publisher[i]id.;
16. For each Publisher Pi;
17. If DataAvailable for topic (Ti);
18. Broker send SubscriberList of Subscriber and Subscribed topic to publisher;
19. For each Si in SubscriberList;
20. CipherText $\leftarrow$ Encrypt (Message, Pi, Si);
21. Send CipherText to Si using JSON object on Message Channel;
22. At Subscriber-End Si;
23. Use Key Inventory of that subscriber to acquire CipherTechnique, Key corresponding to Sender of the message;
24. Decrypt the Message.

Algorithm 2: CreateNewKey(sender, receiver)

1. Use the Diffie Hellman key exchange algorithm to generate key parameters and JSON object to encapsulate the key parameters and transfer it through the broker on key channel.

Algorithm 3: Encrypt(message, publisher, subscriber)

1. Use K1 from KeyInventory of Subscriber to get Cipher Technique;
2. Use K2 (and K3: if affine) from KeyInventory of Subscriber and this Cipher Technique to encrypt message;
3. Return Encrypted Message;

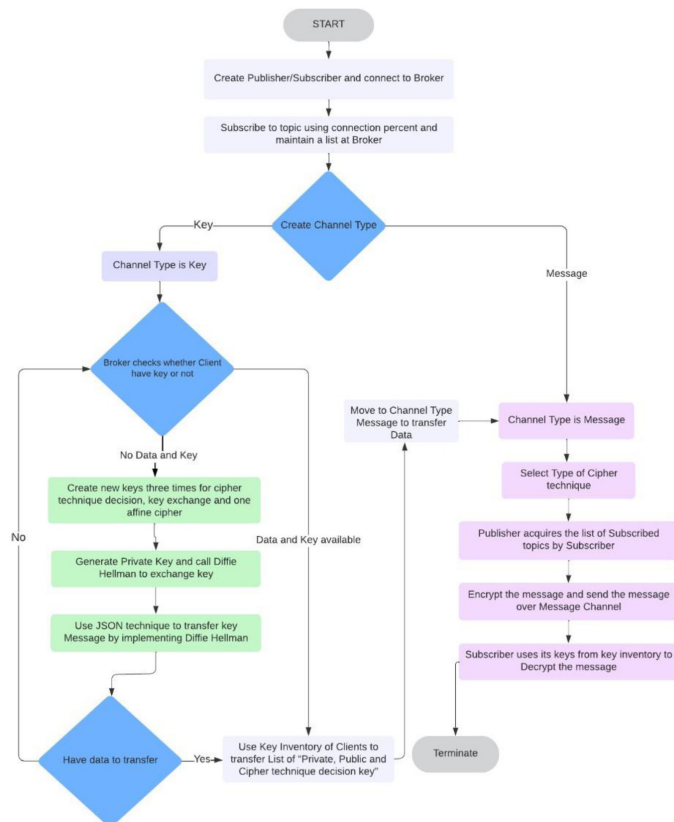


Figure 2

Flow Chart depicting the flow of control for the encrypted message transfer in IoT Networks

6. Result

There is a centralized Broker that is a server in an IoT network whose job is to transfer the data with security between publisher and subscriber. There is a set of publishers who publish data on the Broker over the topics that are registered by them. Publishers can have a sensor that collects data from its environment like temperature, pressure, and humidity in case of Weather Applications. There is a set of subscriber(s) who subscribe to topics which are key storage elements so as soon as the publisher updates anything in the topic they get notified immediately.

6.1 CPU Utilization

From Figure 3, a case where 10 Subscribers were created using an average of 7 percent of CPU processing power, and when the subscriber increased by 10 times the average processing power was increased by 200 percent. This implies that the number of subscribers is not linearly proportional to the processing power required for simulation.

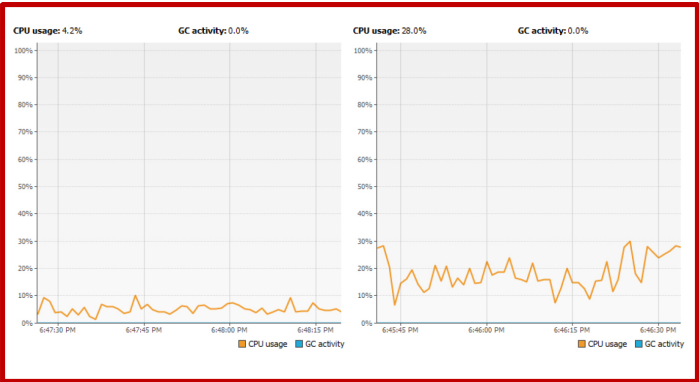


Figure 3
CPU Utilisation of 10 vs 100 Subscribers

6.2 Memory Utilization

When checked for memory consumption (Figure 4), only 45 percent increase was registered in 100 subscribers implementation against 10

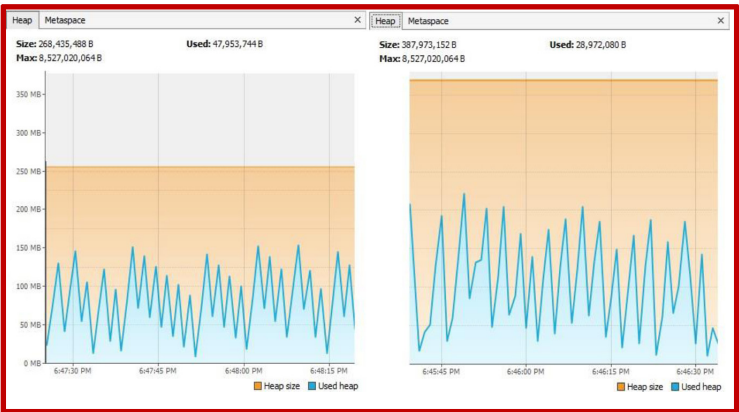


Figure 4
Memory Utilisation of 10 Vs 100 Subscribers

subscribers implementation. It means that an increment in the number of subscribers does not impact memory consumption by that much.

7. Conclusion and Future Work

This seems to be the first of its kind, approach for transmitting the encrypted message as well as keys from publisher to subscriber using IoT Broker. Diffie Hellman Algorithm is used for transmitting keys along with the encoded cipher technique. Key Inventory is used to share and store the keys and acquire the sender ID of that key at the receiving end. All this simulation is done under some specific software and hardware specifications. One can improve or change the specification of the system and can change the number of clients and topics to see the effect on CPU utilization, memory utilization, and execution time.

Acknowledgment

The author(s) are thankful to the research-oriented academic environment provided by the GGSIP University and also thankful to AICTE for providing the ADF fellowship (AICTE Doctoral Fellowship). The authors(s) are thankful for the cooperation provided by Mr. Prateek Kumar Rai and Mr. Mohammad Hasim in proofreading and compilation of the results in the paper.

Submission

This research paper is an expanded and extended version of the "SEROMI: Secured Encrypted Routing of Message in IoT and Crypto-Economic Model for Data Security in IoT Network" [10] [11].

References

- [1] Rekha, Shashi, et al. "Study of security issues and solutions in Internet of Things (IoT)." *Materials Today: Proceedings* 80 : 3554-3559 (2023).
- [2] Chaudhary, Sapna, et al. "CRAIoT: concept, review and application (s) of IoT." 2019 4th international conference on internet of things: Smart innovation and usages (IoT-SIU). IEEE (2019).
- [3] Johari, Rahul, Shubhi Bansal, and Kanika Gupta. "Routing in IoT using MQTT protocol." 2020 12th International Conference on Com-

- putational Intelligence and Communication Networks (CICN). IEEE (2020).
- [4] Kumar, Vishal, Gayatri Sakya, and Chandra Shankar. "WSN and IoT based smart city model using the MQTT protocol." *Journal of Discrete Mathematical Sciences and Cryptography* 22.8 : 1423-1434 (2019).
 - [5] Limkar, Suresh, et al. "A mechanism to ensure identity-based anonymity and authentication for IoT infrastructure using cryptography." *Journal of Discrete Mathematical Sciences and Cryptography* 26.5: 1597-1611 (2023).
 - [6] Haseeb, Khalid, et al. "Trust management with fault-tolerant supervised routing for smart cities using internet of things." *IEEE Internet of Things Journal* 9.22 : 22608-22617 (2022).
 - [7] Dankan Gowda, V., et al. "Vector space modelling-based intelligent binary image encryption for secure communication." *Journal of Discrete Mathematical Sciences and Cryptography* 25.4 : 1157-1171 (2022).
 - [8] MQ Telemetry Transport (MQTT) V3.1 Protocol Specification, <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html>.
 - [9] Su, Wei-Tsung, Wei-Cheng Chen, and Chao-Chun Chen. "An extensible and transparent thing-to-thing security enhancement for MQTT protocol in IoT environment." 2019 Global IoT Summit (GloTS). IEEE (2019).
 - [10] Sonam, and Rahul Johari. "SEROMI: Secured Encrypted Routing of Message in IoT." International Conference on Advanced Communications and Machine Intelligence. Singapore: Springer Nature Singapore (2022).
 - [11] Sonam, and Rahul Johari. "Crypto-Economic Model for Data Security in IoT Network." International Conference on Innovative Computing and Communications: Proceedings of ICICC 2022, Volume 3. Singapore: Springer Nature Singapore (2022).