ISBN 978-981-11-0008-6
Proceedings of 2016 6th International Workshop on Computer Science and Engineering (WCSE 2016)
Tokyo, Japan, 17-19 June, 2016, pp. 340-344
doi: 10.18178/wcse.2016.06.053

An Adaptive Energy-efficient SSL/TLS Method for the Internet of Things using MQTT on Wireless Networks

Jin Hee Chung and Tae Ho Cho [+]

College of Information and Communication Engineering, Sungkyunkwan University, Suwon 16419, Republic of Korea

Abstract. Internet of Things (IoT) devices use MQTT, which is a lightweight messaging protocol, with SSL/TLS for security communication. The original SSL/TLS selects a cipher suite that has the highest preference at the negotiation step of a handshake. This selected cipher suite provides a high security level. However, ensuring secure communication can be intensive for IoT devices when providing much higher security than the expected security level. This limitation causes wasting excessive energy use and overhead in the device. In this paper, we propose an adaptive energy-efficient SSL/TLS for IoT devices using MQTT with SSL/TLS. Our proposed method provides suitable security communication that adapts to the environment of the device. The experimental results demonstrate the validity of our proposed method, which leads to an energy savings of 34.72%.

Keywords: SSL/TLS, Internet of Things, message queuing telemetry transport, security.

1. Introduction

The Internet of Things (IoT) is an infrastructure connecting many different kinds of intelligent objects [1, 2]. These objects are not only computers but could also be a pen that can connect to the Internet. The IoT brings many benefits to our life and contributes to the development of society. However, security problems related to the IoT have become more pressing as the number of security accidents have increased. One of the many different causes for these IoT security vulnerabilities is that IoT devices have low performance; this is the case because most IoT devices use low-energy and low-performance hardware to reduce unit costs. Therefore, due to this lower overhead, a comparatively lower security level is obtained.

IoT devices use message queueing telemetry transport (MQTT), which is a lightweight IoT standard protocol, due to the constrained performance. MQTT sends messages in plaintext; thus, it relies on SSL/TLS to tighten security [3, 4]. However, the handshake of SSL/TLS can be quite intensive for constrained devices [5]. Additionally, it causes devices that are not constrained to become exhausted. There are public purpose and commercial purpose communications that spend supernumerary energy on the handshake and are used just once. Alternatively, there are personal purpose communications that communicate frequently and use reconnects; these spend much less energy once they have a full handshake. Devices use energy inefficiently as the number of disposable handshakes increases. Full handshakes of SSL/TLS lead to energy waste and processor overhead. Statically, overhead of the full handshake is 6.5 KB; thus, this value can represent significant overhead for constrained devices [6]. In addition, IoT devices that use SSL/TLS have another problem. Even if the device can handshake sufficiently, the energy is limited and the security communication should be achieved with low residual energy. These devices need to spend energy efficiently in order to have more handshakes and maintain longer lifetimes; however, if they use SSL/TLS, they can use more energy than is necessary to provide a high security level. For instance, some devices waste energy to protect data that do not require a high security level (e.g., notices and advertisements).

[+] Corresponding author. Tel.: + 82-31-290-7221.
E-mail address: {jinhee91, thcho}@skku.edu.

In this paper, we propose an energy-efficient SSL/TLS method for the IoT to help security communications. The background of the proposed method is explained in Section 2. In Section 3, we elaborate on our proposed method using an evaluation function, which has three inputs (the security level, purpose of communication, and residual energy), and a dynamic cipher suite decision algorithm. In Section 4, we describe the experimental results. The final section explains our conclusions and future work.

2. Background

We discuss MQTT and SSL/TLS in this section.

2.1. MQTT

MQTT is a lightweight IoT standard messaging protocol accepted by OASIS (organization for the advancement of structured information standards) [7].

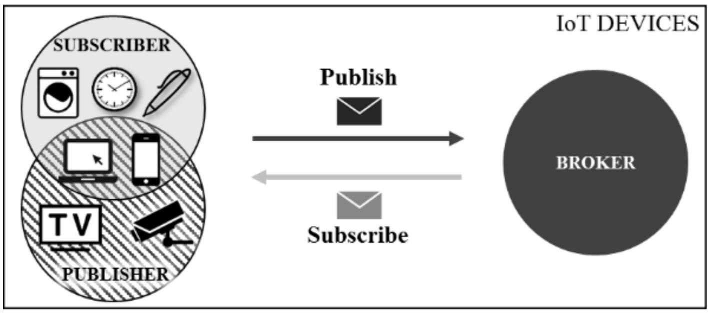

Fig. 1: The process of MQTT

Fig. 1 shows the process of MQTT. MQTT distinguishes IoT devices into broker, publisher, and subscriber components. The broker is an intermediary between the publisher and the subscriber. The publisher publishes a message to the broker, and the subscriber subscribes to some topics and takes messages related to those topics. MQTT does not use encrypted communication; thus, it relies on SSL/TLS.

2.2. SSL/TLS

SSL/TLS is a proposed IETF standard and is used on the transport and application layers [8]. The first step of the handshake is to send the client hello message to negotiate the cipher algorithms that are used in the session. A client sends a list, which is referred to as the cipher suites; this is organized in the preference order of the client. Also, there are many cipher suites, which detail the methods related to the security communication; this data includes the key exchange, authentication, encryption, and MAC. The server chooses the cipher suite with the highest preference that can be provided by the server and client. After the negotiation is complete, the chosen cipher suite is used throughout the session and the security communication is started.

3. Proposed Method

In this section, we elaborate on our proposed method by using an evaluation function.

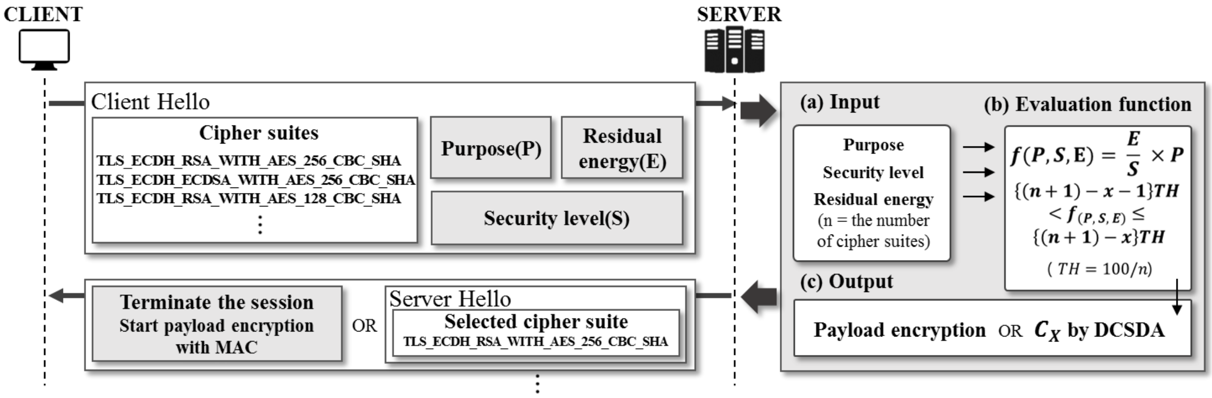
CLIENT
SERVER
Client Hello
Cipher suites
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
Purpose(P)
Residual energy(E)
Security level(S)
Terminate the session
Start payload encryption with MAC
OR
Server Hello
Selected cipher suite
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
(a) Input
(b) Evaluation function
Purpose
Security level
Residual energy
(n = the number of cipher suites)
$f(P,S,E) = \frac{E}{S} \times P$
$\{(n+1)-x-1\}TH < f_{(P,S,E)} \leq \{(n+1)-x\}TH$
$(TH = 100/n)$
(c) Output
Payload encryption OR C_X by DCSDA

Fig. 2: The proposed method

3.1. An adaptive energy-efficient SSL/TLS method

Fig. 2 presents our proposed method. In our proposed method, the client (a subscriber or a publisher of an IoT device) sends the client hello message to the server (which can be a broker) with three additional pieces of information (Fig. 2 (a)). The additional information includes 1) whether the purpose of this communication is personal or public, 2) the security level that is expected, and 3) the residual energy of the client device. The server calculates an evaluation function (Fig. 2 (b)), and the results of this function indicate the device's ability to communicate using SSL/TLS. After this calculation, our forward direction is divided into two methods (Fig. 2 (c)). First, only the payload of MQTT is encrypted and verified with MAC. Second, one cipher suite is selected by the dynamic cipher suite decision algorithm. In Section 3.2, we explain the evaluation function specifically. In Section 3.3, we elaborate on the dynamic cipher suite decision algorithm.

3.2. Evaluation function

Equation (1) is the evaluation function used in the proposed method. It estimates the ability of a device to communicate using SSL/TLS. First, if the purpose of the communication is not personal, then it regards this handshake as supernumerary overhead, encrypts the payload, and carries out verification with MAC. Alternatively, if the purpose is personal, it applies to the dynamic cipher suite decision algorithm using the results of the evaluation function as an argument. The dynamic cipher suite decision algorithm is described in Section 3.3.

$$f(P,S,E) = \frac{Residual\ Energy\ (E)}{Security\ Level\ (S)} \times Purpose\ (p) \tag{1}$$

3.3. Dynamic cipher suite decision algorithm

This algorithm calculates a threshold and establishes a range for each cipher suite. Next, if the results of the evaluation function are in a certain range, the cipher suite that represents the range is selected. The method used to specify the range and the cipher suite is described by Table 1 and (2). First, the cipher suites are rearranged based on their scores. To explain scoring, we use the cipher suites that were used in the experiment (Table 2). As the energy consumption decrease and preference of a cipher suite increase, its score increases. Also, each cipher suite has an additional score related to the security level, which is supported by algorithms that provide confidentiality and integrity. For instance, AES_256_CBC obtains a higher score than AES_128_CBC in terms of the confidentiality. The purpose of scoring is to keep the security as high as possible and to find the most suitable cipher suite for the client's situation. After scoring, the order of the cipher suites is rearranged in descending order. If the total score is the same, then the cipher suite that has the highest cipher preference acquires priority.

Table 1: Cipher suite evaluation score

Score / *C*	*Energy consumption*	*Client preference*	*+ Encryption algorithm*	*+ MAC algorithm*	*Rearranged order*
C1	1	6	7	8	**3**
C2	2	5	7	8	**2**
C3	3	2	5	5	**6**
C4	4	3	7	7	**5**
C5	5	4	9	10	**1**
C6	6	1	7	7	**4**

Additionally, it is necessary to establish ranges and select one cipher suite based on the rearranged order. When the number of cipher suites is n, variable *TH* (threshold) is *100/n*. If the rearranged order in Table 1 is x, then the range is as follows:

$$\{(n+1)-x-1\}TH \ < \ f(P,S,E) \ \leq \ \{(n+1)-x\}TH \tag{2}$$

This compares the calculated values of the evaluation function, determines the corresponding cipher suite, and carries out communication using the decided cipher suite. For instance, if the value of the evaluation function is between *{(n+1) -4 -1}TH* and *{(n+1) -4}TH*, then $Cipher\ suite_4$ is selected. This means that C6 is selected, which is fourth in the rearranged order. However, if the value of the evaluation function is lower or the same as *{(n+1) -n}TH/2*, then it is determined that this device does not have the ability to have an SSL/TLS handshake; only the payload is encrypted and verified with MAC, similar to the public purpose case, despite the fact that the purpose is personal.

4. Experimental Results

In this section, we describe the initial parameters of the experiment and provide the experimental results.

4.1. Initial parameters

Table 2: Initial parameters

No.	***Cipher suite***
C1	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
C2	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA
C3	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
C4	TLS_DH_RSA_WITH_AES_128_CBC_SHA
C5	TLS_RSA_WITH_AES_256_CBC_SHA
C6	TLS_RSA_WITH_AES_128_CBC_SHA

For the experiment, we determine the cipher suites that can be provided to both the server and the client, as shown in Table 2. The length of the transmitted message and the residual energy are generated randomly for each trial. Moreover, the number of experiments, which compare the original SSL/TLS and proposed SSL/TLS, was five hundred. The consumption energy of a cipher suite refers to other papers [9]. Also, the preference of the cipher suite refers to OPEN SSL version 1.0.2.

4.2. Results

Fig. 3 and Fig. 4 show the results. The original SSL/TLS selects the cipher suite (C1) that has the highest preference. This is done every time and we assume the cipher suites that are described in Table 2. We compare the energy consumption of the selected cipher suite, the payload encryption with MAC, and the energy consumption of the original SSL/TLS. Consequentially, the proposed method consumes much less energy; the energy usage is reduced by 34.72%. The reason for this improvement is that the proposed method selects two directions to provide security communication and determines the appropriate cipher suite for a diverse array of situations. It significantly increases the gap using payload encryption, and the security communication with the cipher suite selected by the dynamic cipher suite decision algorithm (DCSDA) decreases the energy usage by 21.92% (Fig. 4).

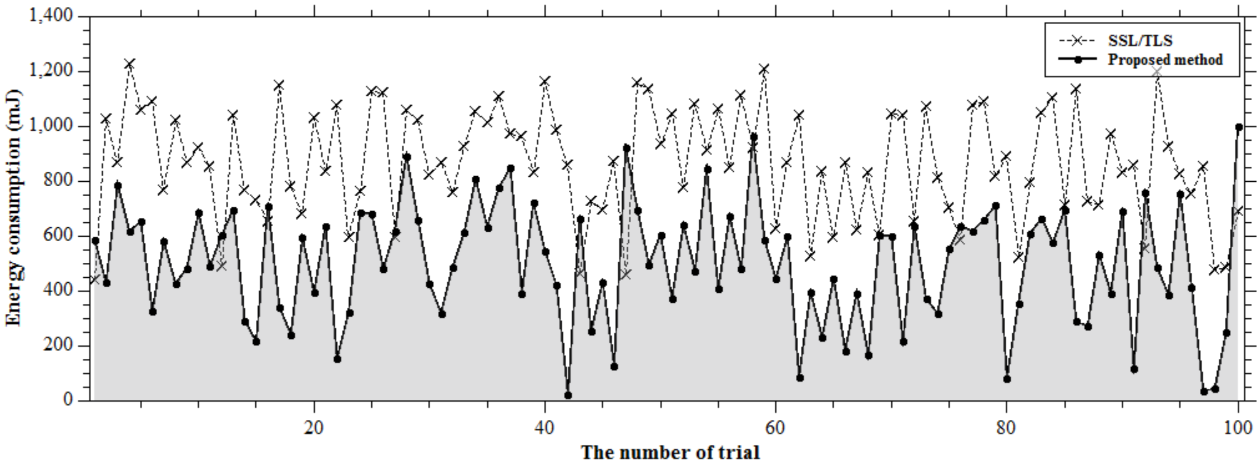

Fig. 3: Experimental result

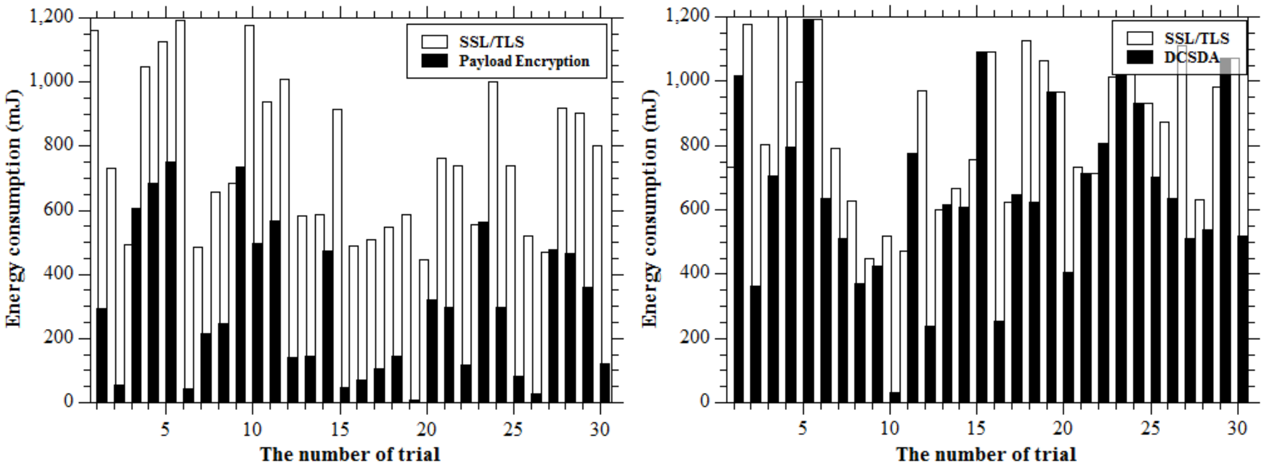

Fig. 4: The results of payload encryption and DCSDA

5. Conclusions and Future Work

MQTT recommends that IoT devices using MQTT use SSL/TLS. However, using original SSL/TLS can be intensive for IoT devices, and it does not consider constrained environments; thus, it can use energy inefficiently. This paper proposes a method using payload encryption, an evaluation function that has three inputs (the communication purpose, expected security level, and residual energy), and a dynamic cipher suite algorithm during the first step of the handshake. It facilitates security communication and enhances the energy efficiency by 34.72%, as described in Section 4. In order to further improve our proposed method, we plan to consider the vulnerability when encrypting only the payload with MAC; we hope to compensate for this defect in our future works.

6. Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2015R1D1A1A01059484).

7. References

[1] K. Zhao and L. Ge, “A survey on the internet of things security,” in *Computational Intelligence and Security (CIS), 2013 9th International Conference On*, 2013, pp. 663-667.

[2] S. Oteafy and H. S. Hassanein, “Resource re-use in wireless sensor networks: Realizing a synergetic internet of things,” Journal of Communications, vol. 7, pp. 484-493, 2012.

[3] I. Ishaq, D. Carels, G. K. Teklemariam, J. Hoebeke, F. V. d. Abeele, E. D. Poorter, I. Moerman and P. Demeester, “IETF standardization in the field of the internet of things (IoT): a survey,” *Journal of Sensor and Actuator Networks*, vol. 2, pp. 235-287, 2013.

[4] S. Bandyopadhyay and A. Bhattacharyya, “Lightweight internet protocols for web enablement of sensors using constrained gateway devices,” in *Computing, Networking and Communications (ICNC), 2013 International Conference On*, 2013, pp. 334-340.

[5] Hive MQ, “MQTT Security Fundamentals: TLS / SSL, “Available: http://www.hivemq.com/blog/mqtt-security-fundamentals-tls-ssl.

[6] Anonymous “TLS overhead,” Available: http://netsekure.org/2010/03/tls-overhead/, MAR 12TH, 2010.

[7] A. Banks and R. Gupta, “MQTT Version 3.1. 1,” *OASIS Standard*, 2014.

[8] ITU-T, “X. 800 security architecture for open systems interconnection for ccitt applications,” 1991.

[9] N. R. Potlapally, S. Ravi, A. Raghunathan and N. K. Jha, “A study of the energy consumption characteristics of cryptographic algorithms and security protocols,” Mobile Computing, IEEE Transactions on, vol. 5, pp. 128-143, 2006.