A Proposal on Improved Data Security Mechanism in Smart Home Applications based on IOT-CoAP Protocol

Dr. K.N. Venkata Ratna Kumar

Professor & Principal - Engineering, St. Mary's Group of Institutions Guntur, Guntur, Andhra Pradesh, India

Abstract - Now-a-days, the progress in technology is highly exponential that made the internet extended to everything such as temperature sensors. These sensors require protocols to send and receive data such as the Constrained Application Protocol (CoAP). Nevertheless, these protocols must be light, therefore, these sensors are restricted with energy. However, these light protocols must also be secure. In this paper, an enhancement on a CoAP protocol is endeavoured by focusing on the message integrity, and the author attempted to find the optimal hash function that can be added to the CoAP protocol in order to increase the security without influencing the performance. Three types of hash function have been considered with the CoAP protocol, which are SHA-1, SHA224, and SHA256. The enhanced protocol has been evaluated using the Contiki OS simulation tool on a smart home application, and the results show that SHA 224 is the best hash algorithm according to the performance.

Keywords - *Ad-hoc network, CoAP, Hash function, integrity.*

I. INTRODUCTION

The Constrained Application Protocol (CoAP) is a protocol that runs over the UDP in IoT [1] that is designed for simplicity which is used in web data transfer and is used with restricted nodes and restricted networks in the internet of things [2]. This protocol is used with machine-to-machine (M2M) connection [3] such as smart home applications or smart parking system, etc.

CoAP protocol is acting as HTTP protocol in the normal network [3], and as it is known that HTTP protocol is not a secure protocol [4]. Therefore HTTP protocol is used with other protocols such as TLS [5], SSH [6] or IPsec [7] to secure the HTTP protocol. The same applies to CoAP protocol as it could be used with other protocols such as DTLS to secure it [8].

All of them don't fit for communication between IoT devices but a few protocols have been used for communicating IoT devices based on their light weightiness in the architecture. Some protocols such as CoAP, MQTT, AMQP, XMPP, DDS [9] deliver better power efficiency and reliability for IoT needs. Based on the applications used in daily life, protocols are selected. In this paper, home automation is implemented by using CoAP protocol to transfer data among different IoT devices within the same network. Constrained Application Protocol (CoAP) has UDP connection in transport layer while rest of them have TCP connection in their transport layer. It is equipped with DTLS security [1] [11] and allows confirmable and non-confirmable messages in process of transferring data. Data can be monitored from smart phones and laptops by connecting to that network. Message Queuing Transportation Telemetry (MQTT) protocol is widely spread and mostly used, but it has its own cons. Some of them have been overcome by using CoAP protocol [10].

IoT CoAP protocol is an application layer protocol and is predestined to be the protocol most commonly used with all application protocols [12]. CoAP is a new version of HTTP and runs over UDP for the transactions. Unlike HTTP which runs over TCP and it is inappropriate for use in the constrained environments [13]. CoAP is designed to be lightweight and supports resource-constrained environments [14].

Datagram Transport Layer Security (DTLS) is proposed to be used with CoAP protocol as the security responsible for the data encryption and integrity protection, key management, and authentication. Similar to HTTPS protocol, CoAP with DTLS is termed secure CoAP (CoAPS) [15]. However, DTLS protocol is a heavy protocol for the restricted IoT devices in terms of time and energy consumption, because it requires many message exchanges to initiate a secure session [16].

Connecting everyday objects to the Internet will involve the use of embedded devices (Cormann et al., 2014). These embedded devices use sensors to collect data and wireless radios to achieve Internet connectivity. However, the embedded devices projected to make up the IoT ecosystem bring along their own challenges. There is a need to keep the size of these devices as small as possible to not affect the weight or look of the everyday objects in which they are being implemented. Additionally, to allow mass deployment of IoT devices, businesses want the cost per unit to be as inexpensive as possible. To meet these goals, embedded devices are being built with limited processing power, memory, and energy capacity compared to unconstrained, traditional computing devices (laptops, smart phones etc.). Due to these limitations, they are known as constrained devices.

In application layer, the CoAP is mainly used for secure communication between the constraint smart IoT devices and server because MQTT protocol [18] has some limitations such as it can be used only for very low

processor devices and can communicate mainly for Amazon cloud applications for server [19]. CoAP uses RESTful architecture [17] to access the resources from server through URI(Universal Resource Identifier) and message communication. Thus, CoAP architecture is divided into two layers : (1) message layer and (2) request/response layer. The message layer is responsible for controlling the exchange of messages between devices over UDP (User Datagram Protocol). The request / response layer is responsible for handling the requests of IoT devices and corresponding responses from other devices/server through message communication and also maintains the status of the messages like out of order, lost or duplicated etc. The request/response layer is also responsible for manipulating the resources by using one of the various transmission methods such as GET, PUT, POST and DELETE.

II. LITERATURE REVIEW

Authors in [20] proposed a "Lithe - an integration of DTLS and CoAP for the IoT", in their mechanism, they compressed the DTLS Header, they did not aim to secure the transition rather than making the DTLS protocol more compatible with restricted IoT devices through reducing the power consumption by minimizing the number of transmitted bytes. In their work, they used the Contiki operating system for evaluating the proposed mechanism including the packet size, the performance of the proposed mechanism, and the consumption of the energy, and the results showed significant gains.

In [21], the authors encrypted the transmission by using the IPsec protocol between the sensor network and the traditional internet in order to encapsulate the transmission, their approach adopted the Authentication header (AH) protocol and Encapsulation Security Payload (ESP) protocol under IPv6 in order to encrypt, authenticate and check the message integrity.

It appears that much research into securing CoAP protocol has been proposed, designed and implemented. Reference [22], proposes a security scheme using the RSA algorithm.

Perelman (Perelman, 2012, pp. 37-41) created his own TLS and DTLS implementations for Contiki OS 2.6/2.7 and was able to run a memory analysis of static RAM and flash ROM usage. In addition, he mapped out time requirements for the DTLS handshake. However, his implementation is no longer supported by the latest version of Contiki and it was never tested with CoAP.

In "Lithe: Lightweight Secure CoAP for the Internet of Things", Raza et al. (2013) further presented a quantitative 3 evaluation of their model and demonstrated that their DTLS compression scheme enabled a lower energy consumption and processing time as compared to the uncompressed version. In a Java based implementation of CoAP-DTLS, Stefan Jucker (2012) found that the round-trip time of a message sent between two traditional computers increases by over four times for CoAP-DTLS messages compared to just CoAP messages. While all these research works have provided great insights about performance differences between secure, light-weight protocols, their proposed models have used simulations and tested protocols using unconstrained devices and environments.

In 2018, Albalas et al. [23] presented the performance evaluation between the CoAP using ECC and CoAP using RSA based on three different factors—*(i) message length, (ii) security services and (iii) residual energy*. It is mentioned that the CoAP using the ECC is 47% efficient in saving energy than CoAP using RSA due to smaller key size of ECC.However, the CoAP using ECC is still facing some problems related to multicasting, asynchronous data communication and key management issues related to CoAP [23]. This study has motivated us to develop ECC-CoAP protocol eliminating most of the aforementioned limitations. In 2017, Harish et al. [24] establishes a secure connection using HTTP between IoT nodes and handles the HTTP request through a proxy which is referred to 6LoBR and maintains the security issues for the CoAP layer by encrypting/decrypting the payload of the corresponding CoAP request/response using ECC. It is managed by an IoT controller which maintains the whole traffic of the wireless network. However, it is found that the scheme [24] is suffering from some key management issues of CoAP.

III. SYSTEM MODEL

In this paper, an enhancement on CoAP protocol is proposed in order to add authentication and integrity to the protocol without using the DTLS, as DTLS it is not extremely efficient, fast and has complex initial handshake. The proposed enhancement will be efficient on performance and lead to less power consumption, besides the cost of implementing this protocol will be less than using the DTLS protocol. The proposed enhancement requires less computational power, as it employs lightweight authentication by embedding a short MAC and using the same message format.

A. Mechanism

The solution will be implemented between wireless sensor and the controller

B. Pre-assumption

In order to make the enhancement on the CoAP protocol, both the sensor and the controller will agree on the following values:

- A sequence-id value in order to avoid the replay attack.
- A pre-shared symmetric key between the sensor and the controller.

• A secure hash function, which is used in a message authentication.

• The compression function, which is LZW.

The following figure represents block diagram of system model :

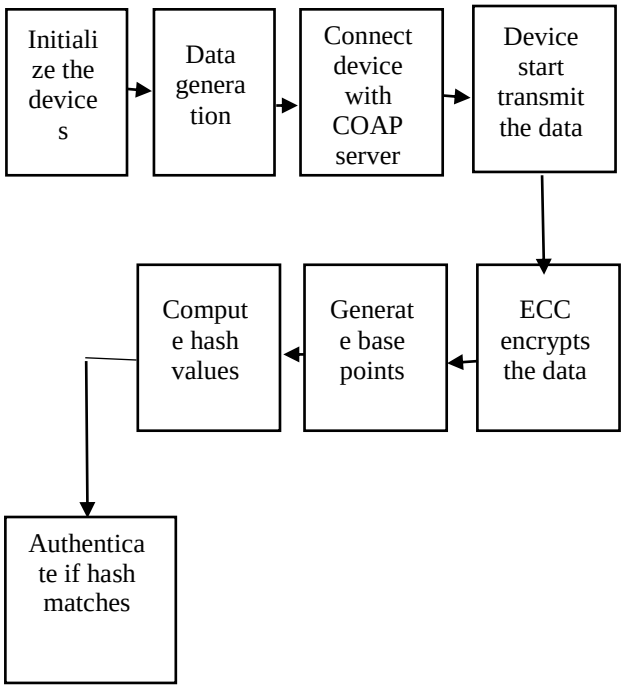

Fig-1: Block diagram of system model

First, the devices are connected with each other using a border router for data transmission / reception to control the IOT devices. The IoT devices are intended to control, monitor the home environments. The devices like air conditioner, smart bulbs and other smart home devices. These devices are controlled and monitored using COAP protocol in IoT applications. COAP is a specialized web transfer protocol for use with constrained nodes and constrained networks in the Internet of Things. CoAP is designed to enable simple, constrained devices to join the IoT even through constrained networks with low bandwidth and low availability. Data integrity is a major concern in IoT COAP as a simple modification in the actual data lead to serious complications in the application. ECC generates a symmetric shared key between the user and server. The key is verified in the server end and the message can be accepted only if the authentication is verified successfully. Due to the randomness of ECC, the intermediate / unknown nodes cannot tamper the messages even if they possess the data packets.

IV. RESULT AND DISCUSSION

Simulation environment:

Instant Contiki is a complete Contiki development environment running within an Ubuntu Linux virtual machine (Ubuntu 14.04 LTS) that has all the compilers, development tools and simulators needed for the study.

V. IMPLEMENTATION AND EVALUATION

In this section, we simulate and evaluate our mitigation scheme on the basis of two parameters : the *aggregate transmission power*, and the *aggregate CPU processing power of the victim node*.

The network topology used in the simulation is shown below :

Fig-2: Initial terminal process

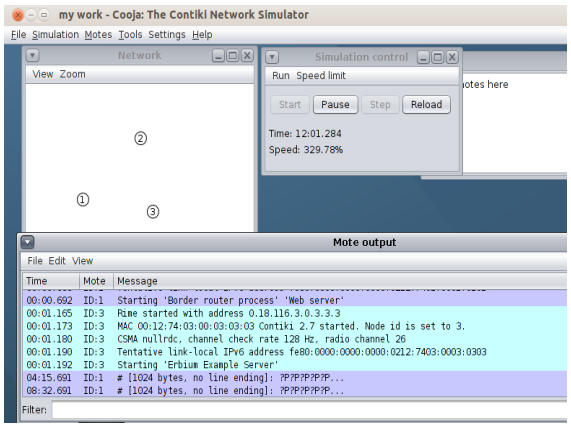

Fig-3: Creation of Motes and start simulation control

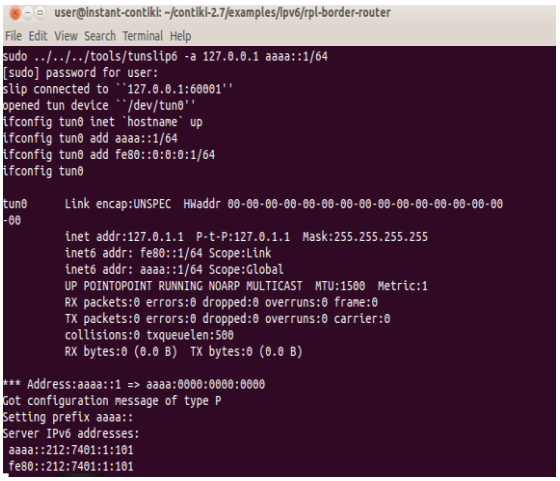

Fig-4: Running the simulation work and IP addresses noticed

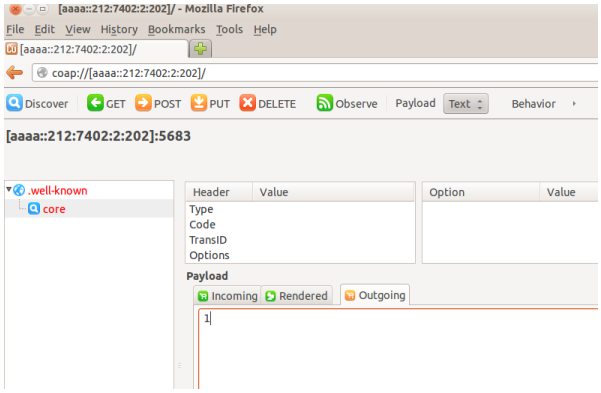

Fig-5: After connectivity to browser

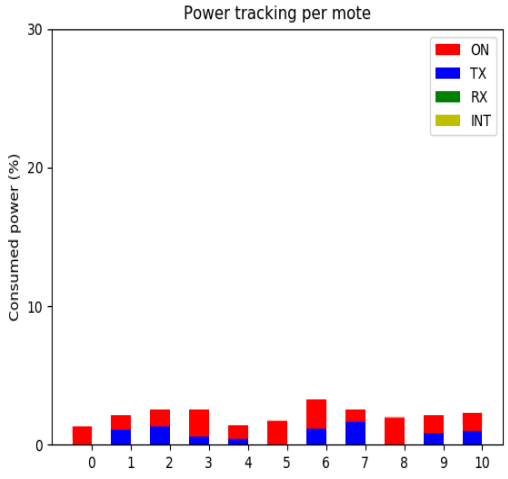

Fig-6: Comparison process of Existing system

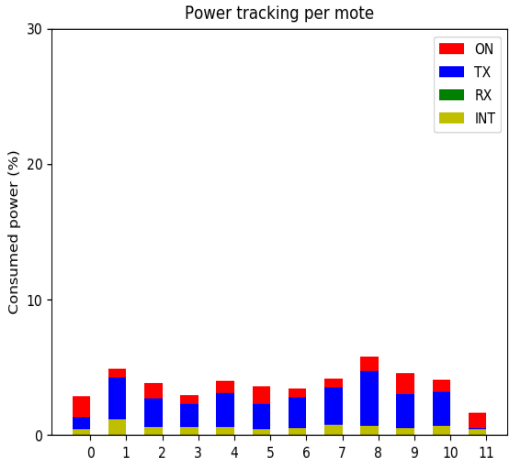

Fig-7: Comparison process of proposed system

VI. CONCLUSION

In this paper, a security enhancement is added to the CoAP protocol by adding the integrity to the CoAP packet. The integrity is achieved by using the hash function; three types of hash functions are implemented and evaluated using Contiki OS with a smart home application. The results show that using SHA224 is the most optimal algorithm with CoAP according to the power consumption and the time execution; this is attributed to the number of rounds in SHA224 as it is 64 but in SHA1 and SHA256 they are 80.

VII. REFERENCES

[1] Kuladinithi, K., Bergmann, O., Pötsch, T., Becker, M., and Görg, C.: 'Implementation of coap and its application in transport logistics', Proc. IP+ SN, Chicago, IL, USA, 2011

[2] Bormann, C., Castellani, A.P., and Shelby, Z.: 'Coap: An application protocol for billions of tiny internet nodes', IEEE Internet Computing, 2012, 16, (2), pp. 62-67

[3] Shelby, C.Z.: 'Constrained application protocol (CoAP', in Editor (Ed.)^(Eds.): 'Book Constrained application protocol (CoAP' (Citeseer, 2011, edn.), pp.

[4] Rescorla, E.: 'SSL and TLS: designing and building secure systems' (Addison-Wesley Reading, 2001. 2001)

[5] Rescorla, E.: 'HTTP Over TLS', in Editor (Ed.)^(Eds.): 'Book HTTP Over TLS' (2000, edn.), pp.

[6] YLONEN, T.: 'The Secure Shell (SSH) Authentication Protocol', RFC 4252, 2006

[7] Stanton, R.: 'Securing vpns: Comparing ssl and ipsec', Computer Fraud & Security, 2005, 2005, (9), pp. 17-19

[8] Peretti, G.: 'CoAP over DTLS TinyOS Implementation and Performance Analysis', MS Thesis, University of Padova, Italy, 2013

[9] Nitin Naik, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP" 2017 IEEE International Systems Engineering Symposium (ISSE).

[10] Chihyun Cho, Jungyong Kim, Younghyun Joo, Jaesick Shin, "An approach for CoAP based notification service in IoT environment" 2016 International Conference on Information and Communication Technology Convergence (ICTC).

[11] Angelo Capossele, Valerio Cervo, Gianluca De Cicco, Chiara Petrioli, "Security as a CoAP resource: An optimized DTLS implementation for the IoT" 2015 IEEE International Conference on Communications (ICC).

[12] N. Naik, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," 2017 IEEE Int. Symp. Syst. Eng. ISSE 2017 - Proc., 2017.

[13] T. Kothmayr, C. Schmitt, W. Hu, M. Brünig, and G. Carle, "Ad Hoc Networks DTLS based security and two-way authentication for the Internet of Things q," Ad Hoc Networks, vol. 11, no. 8, pp. 2710–2723, 2013.

[14] R. A. Rahman and B. Shah, "Security analysis of IoT protocols: A focus in CoAP," 2016 3rd MEC Int. Conf. Big Data Smart City, ICBDSC 2016, pp. 172–178, 2016.

[15] S. Raza, H. Shafagh, K. Hewage, R. Hummen, and T. Voigt, "Lithe: Lightweight secure CoAP for the internet of things," IEEE Sens. J., vol. 13, no. 10, pp. 3711–3720, 2013.

[16] A. Haroon, S. Akram, M. A. Shah, and A. Wahid, "E-lithe: A lightweight secure DTLS for IoT," IEEE Veh. Technol. Conf., vol. 2017–Septe, pp. 1–5, 2018.

[17] Nguyen, H. V., & Iacono, L. L. (2015, September). REST-ful CoAP message authentication. In 2015 international workshop on secure Internet of Things (SIoT) (pp. 35–43). IEEE.

[18] Yassein, M. B., Shatnawi, M. Q., Aljwarneh, S., & Al-Hatmi, R. (2017, May). Internet of Things: Survey and open issues of MQTT protocol. In 2017 international conference on engineering & MIS (ICEMIS) (pp. 1–6). IEEE.

[19] Alliance, O. M. (2002). Generic content download over the air specifcation. v1. 0 December.

[20] Raza, S., Shafagh, H., Hewage, K., Hummen, R., and Voigt, T.: 'Lithe: Lightweight secure CoAP for the internet of things', IEEE Sensors Journal, 2013, 13, (10), pp. 3711-3720

[21] Raza, S., Duquennoy, S., Chung, T., Yazar, D., Voigt, T., and Roedig, U.: 'Securing communication in 6LoWPAN with compressed IPsec', in Editor (Ed.) (Eds.): 'Book Securing communication in 6LoWPAN with compressed IPsec' (IEEE, 2011, edn.), pp. 1-8

[22] T. Kothmayr, C. Schmitt, W. Hu, M. Brünig, and G. Carle, "Ad Hoc Networks DTLS based security and two-way authentication for the Internet of Things q," Ad Hoc Networks, vol. 11, no. 8, pp. 2710–2723, 2013.

[23] Albalas, F., Al-Soud, M., Almomani, O., & Almomani, A. (2018). Security-aware CoAP application layer protocol for the Internet of Things using elliptic-curve cryptography. Power (mw), 1333, 151.

[24] Harish, M., Karthick, R., Rajan, R. M., & Vetriselvi, V. (2018). Securing CoAP through payload encryption: Using elliptic curve cryptography. International Conference on Communications and Cyber Physical Engineering, 2018, 497–511.