Security Schemes for Constrained Application Protocol in IoT: A Precise Survey

Amit Mali(✉) and Anant Nimkar(✉)

Sardar Patel Institute of Technology, Mumbai 400053, India
{amit_mali,anant_nimkar}@spit.ac.in

Abstract. Internet of things is the fast developing network between different day-to-day products or things connected together via Internet. Internet of Things (IoT) has enabled connectivity of millions of devices together and help operate them at ease. The most important factor that needs to be taken into consideration while performing connectivity between devices over IoT is the IPbased communication protocols. Rapid growth in IoT increases security vulnerabilities of the linked objects. Internet Engineering Task Force (IETF) has standardized a communication protocol at application layer, that is developed in consideration with IoT, named Constrained Application Protocol (CoAP). Ensuring security over CoAP is an ongoing challenge and a major research area. CoAP is associated with various security schemes that guarantee secure data transfer and reliability over the network, but each of them still lack in providing full efficiency. This survey aims to analyze different security schemes implied to CoAP inorder to improve its performance and also states issues present in them. We examine different techniques that are aligned with CoAP to ensure fundamental security requirement and protect communication and some research challenges.

1 Introduction

The basic idea behind the Internet of Things (IoT) is connecting various kinds of electronic device into the Internet with an aim to build a worldwide distributed system of interconnected physical objects. For constructing such a global network, where all these nodes should be able to communicate and interact with each other in an efficient manner, software architectures which provide scalability, simplicity and interoperability of communication are required. Due to the unreliable congestion control algorithms, TCP in wireless networks shows a very low performance therefore, the connection-less UDP is mostly used in the IoT. One approach which fulfills these requirements is the architectural style of Representational State Transfer (REST) providing a guideline for designing large-scale distributed applications. The basic idea behind the Internet of Things (IoT) is the integration of all kinds of electronic devices into the Internet with an aim to build a worldwide distributed system of interconnected physical objects.

The security of IoT is a very crucial topic, because it is related to the data that is transmitted over the network, data may be sensitive, personal as well

S.M. Thampi et al. (Eds.): SSCC 2017, CCIS 746, pp. 134–145, 2017.
https://doi.org/10.1007/978-981-10-6898-0_11

as confidential. Protecting these kind of data is a must have in any communication network. According to a research 70% of the ordinarily used IoT devices are found prone to security breach. Some common security complications are insufficient authorization, lack of encryption, and insecure web interfaces [5]. At the end of the day, security, protection and trust are the fundamental components that organizations need to concentrate on while executing IoT environment. Notwithstanding, the greatest test is execution and speed, if security is connected. The IoT gadgets are light and subsequently made remembering low computation power and higher memory capacities in order to perform data transfer between two nodes with minimum delay, and without affecting general throughput avoiding packet loss.

The Constrained Application Protocol (CoAP) is under calibration as an application layer protocol for the IoT [2]. The Constrained Application Protocol (CoAP) designed and maintained by IETF is an application layer protocol constructed mainly for resource constrained devices and M2M applications. It permits data transfer among IoT objects that have UDP and 6lowPAN enabled, achieving low overhead and supporting multicast. CoAP constitutes of two layers, the lower message layer and the upper request/response layer. The message layer provides reliability and sequencing by means of a stopandwait protocol using messages such as confirmable which requires an acknowledgment message as response, non-confirmable which does not require a response, and reset which is used in case a confirmable message cannot be processed [13].

This article analyzes study from various available literature that are present and security techniques for CoAP in the IoT. This survey, presented over in following sections is a legitimate and initial work in this particular domain. Here, the attention is imparted more precisely on the security techniques that help to secure CoAP. Some existing surveys do exist that, focus on the Identification of security requirements but, it is equally important to analyze the security technologies currently being designed for IoT devices [5,6,8].

Our article is organized as follows; Sect. 2 focuses on the Constrained Application Protocol and its Security requirement. In Sect. 3 we discuss various existing security schemes implied to CoAP and Sect. 4 we enlighten some research issues still present in CoAP and we conclude the article on this in Sect. 5.

2 Constarined Application Protocol

The Constrained Application Protocol (CoAP) as its name suggests is developed specifically for IoT networks. Addressing the issue of constrained resources in IoT, CoAP has its dedicated focus on nodes performing data transfer in constrained networks. CoAP is a version of HTTP designed to support requirements for IoT. CoAP depends on Representational State Transfer (REST), a principle adopted from HTTP and embedded in UDP for the transaction [17]. Constrained application Protocol extends its support to provides M2M communication in constrained environments whereas, it also enables optional support uni-cast and multicast requests. Some other notable features depicted by CoAP

are asynchronous message exchanges, low header overhead and parsing complexity, supports URI (Universal Resource Identifier) and content-type, also that it has simple proxy and caching capabilities [3].

2.1 CoAP Architecture

The structure of CoAP is divides into two layer, the message layer and request response layer. The principal layer is in charge of controlling the message trade over UDP between two nodes. While the second layer conveys the request/response which holds respective code with a specific end goal to maintain message delivery, for example, the entry of messages that are out of request, lost or copied. Figure 1 illustrates the design for CoAP. CoAP is a solid instrument with rich components, for example, basic stop-and-wait re-transmissions, copy discovery and multicast support. CoAP utilizes a short fixed-length binary header and components, and messages are encoded in binary simple format. The techniques supported in CoAP depend on the REST-ful structure which is GET, POST, PUT and DELETE [3].

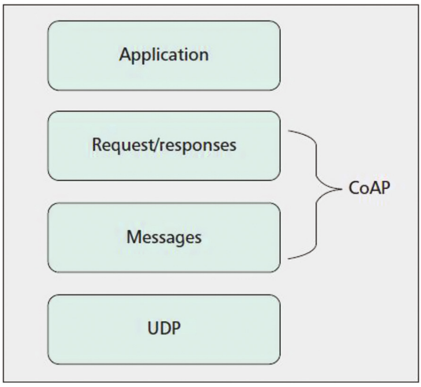

Fig. 1. Architecture of CoAP

CoAP message format is shown in Fig. 2. The CoAP start header contains a version number (V), a message type number (T), a token length (TKL), a code (C) and a Message ID (MID). Since CoAP uses the unreliable UDP, senders can advise receivers to confirm the reception of a message by declaring it as confirmable. The TKL field defines the size of the token which enables the asynchronous message exchange. Based on this token, requests and responses can be matched. The CoAP start header concludes with a Message ID being an identifier for linking a reset or an acknowledgement message to its confirmable message. The next element of the CoAP header is the token value. This value can be

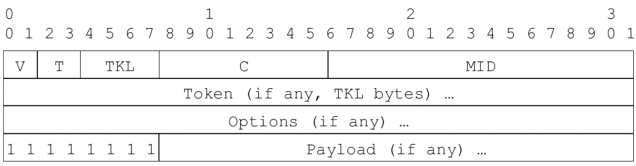

Fig. 2. CoAP message format

empty, if no asynchronous message exchange is needed. The CoAP options complete the CoAP header. The delimiter to separate the header from the payload is a 8-bit unsigned integer with the fixed value of 255.

2.2 Security in CoAP

CoAP, as said, lacks built-in security mechanism and hence, security for CoAP requires the presence of an external security scheme or mechanism, e.g., HTTP and Transport Layer Security (TLS) [8]. As widely used and mentioned by IETF and CoRE working group, security considerations are implemented by using Datagram Transport Layer Security (DTLS) or IPSec [1]. DTLS ensures features such as confidentiality, integrity, authentication, and non-repudiation in the network using AES/CCM. DTLS is mainly functional in the transport layer of the protocol stack. DTLS was at first intended for traditional networks, but over the time it has been ported for constrained devices but this result in producing a heavyweight protocol. DTLS headers are likewise too long to fit in a single IEEE 802.15.4 maximum transmission unit (MTU) [15]. Calculation overhead of their DTLS handshake presents high vitality utilization because of the utilization of RSA-based cryptography.

DTLS is a derived protocol, obtained by modification of Transport Layer Security (TLS) protocol, and it is implied at application layer. DTLS contains records that are 8 bytes longer than in TLS. 13 bytes extra overhead per datagram is incurred on DTLS after the handshake is processed making it costly for constrained nodes [13]. For an incoming message during handshake, it will be decompressed and decryption will be performed by the protocol to verify it. While in an outgoing scenario of handshake, the protocol will apply encryption algorithm, add authentication code (MAC) and compress the message. Following Fig. 3 states the DTLS handshake mechanism between a client and a server.

The security in CoAP is still under talk, despite the fact that DTLS is joined as an assurance layer. The open deliberation is the substantial cost of computation and high handshake in the message which causes message discontinuity. Many reviews have proposed an answer for compressed DTLS which is addressed in further sections. Moreover, key administration is another downside of the CoAP security which is a typical issue in all protocols. Raza et al. have proposed to receive 6LoWPAN header size reduction for DTLS [13]. They have connected compressed DTLS with the 6LoWPAN standard, accomplishing an enormous lessening in the quantity of extra security bits.

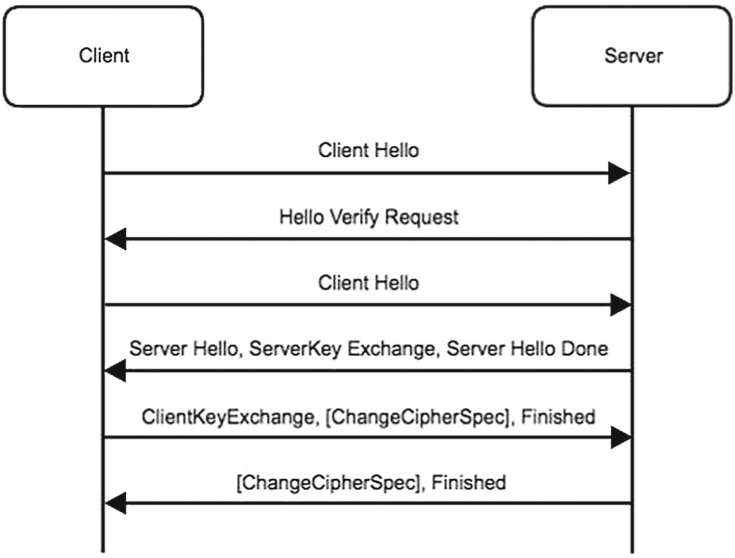

Fig. 3. Handshake mechanism using DTLS

Research is also carried out in order to introduce a symmetric key-based, cost effective security mechanism using authentication and confidentiality for CoAP [19]. Here symmetric key is used with Advanced Encryption Standard (AES) 128 Cipher Block Chaining (CBC) mode. Having a different perspective Oscar et al. [4] have also proposed a method based on new variant of Host Identity protocol that uses pre-shared keys (PSK) and uses AMIKEY protocol for key management. It isn't a standard yet but is definitely reliable. These are some of the security mechanisms implied to CoAP. Furthermore, we will study many such security mechanisms put forth by researchers and scholars aiming towards security to CoAP.

3 Existing Security Scheme for CoAP

This section allows us to study about various security techniques that are aligned along with CoAP to render security. Each of this technique demonstrates its unique feature to attain secure data transfer and reliability in IoT environment over CoAP. We will hereby study each of this technique, their strategy to for securing CoAP and issues present in this technique.

3.1 Security Using DTLS

Datagram Transport Layer Security (DTLS) is primarily aligned as a security protocol with Constrained Application Protocol (CoAP) for specified facilities

such as automatic key management, data encryption and authentication [2]. Secure-CoAP (CoAPs) is a collaborative term including CoAP and DTLS support. Firstly, DTLS was developed and framed for traditional networks and not for IoT devices that posses constrained environments. As Maximum Transmission Unit (MTU) for 802.15.4 is 128 bytes, hence there is a need to compress the DTLS headers and messages. Raza et al. in 2012 firstly proposed a lightweight DTLS support for the IoT using 6LoWPAN header compression standards. 6LoWPAN has a plug-in 6LoWPAN-GHC [14] which is used to compress UDP payload. DTLS is similarly compressed using these standards. 6LoWPAN-GHC allows us to compress record header, handshake header and other handshake messages efficiently that can reduce the packet size and improve the memory consumption.

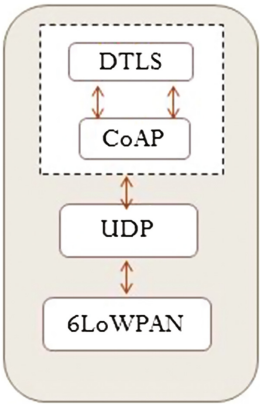

Fig. 4. CoAP DTLS interaction

Further in 2013, Raza et al. devised a security scheme Lithe, which is a lightweight security solution for CoAP that uses 6LoWPAN header compression technique to compress DTLS in order to implement it as security support for CoAP [14]. It is a novel method in all aspects for securing CoAP over the Internet of Things. Evaluation results of this technique over simulation environment in Contiki OS proved to device some positive results that showcased very less amount of bytes transferred resulting in an efficient and CoAP implementation. The header compression reduces a huge amount of traffic in the network, leading to minimal energy consumption. Auther here promises to obtain around 62% of space saving due to compression in comparison with uncompressed ones. It is also observed that in handshake phase compressed DTLS header archives space saving upto 75%. In comparison to plain CoAP, the response time was drastically reduced which proves DTLS compression efficient in terms of energy consumption. Also Lithe avoids fragmentation which results in fragmentation attacks over an IoT system.

To implement Secure CoAP, Park et al. proposed a technique according to which a handshake between a client and a sever will be divides using the secure service manager (SSM) into handshake phase and encryption phase [12]. This result to overcome various problems in LLN such as data loss, delay that contribute in increasing the overhead in network. This separation also prevents the system from DoS attacks as encryption phase is separated at host location. User has a choice of selecting multiple numbers of cipher suites while using this proposed system. Separation of DTLS protocol into a handshake phase and an encryption phase does not have a effect end-to-end security as data encryption and decryption are done in the end node. This system is resistant to SSM spoofing attack, Single point of failure, fragmentation attack and DoS attacks on a constrained device. This system is allows usage of pre-shared key that enables to maintain the relation of the SSM and constrained devices as a single system virtually even if the are physically distributed.

3.2 Security Using Key Management

As an replacement for heavy DTLS, key based authentication technique can be used for securing CoAP messages. Certain methods are proposed that state benefits of Key based authentication and also have been effective when compared with DTLS based conventional approach. Ukil et al. proposed a security solution which is based on symmetric key [20]. Exchanged symmetric key is used with Advanced Encryption Standard (AES) 128 Cipher Block Chaining (CBC) mode. This method consists of phases such as secret distribution; session initiation; server challenge; sensor response. Being an payload embedded method supports towards minimizing the handshake overhead.

Furthermore Bandopadhy et al. came up with Auth-Lite [19] that enables security to CoAP by providing object security. Security is ensured using a technique where key management and authentication scheme are integrated together and are based upon usage of Symmetric keys. Resource consumption in CoAP is minimized by introducing new header options along with mutual authentication. Auth-Lite is manned to protect the system threats of DoS attack, replay attack, man-in-the-middle attack, and information disclosure attack. When evaluated against DTLS based CoAP system it is observed that Auth-Lite has higher performance and less losses in a pre-shared key mode. Auth- Lite combined with DTLS provides a perfect security solution that provides mutual authentication layer and also protect from various attacks.

Obtaining all-round security in constrained environment is difficult as the existing network security protocols lack to provide support for all required functionalities and traditional Internet solutions provide deprived performance when implied to constrained devices To overcome this situation a new variant of Host Identity Protocol (HIP) based on Pre- Shared Key (PSK) is stated which introduces a cryptographic namespace of stable host identifiers between network and transport layer to improve the performance and reliability of constrained networks. Oscar et al. [4] propose a solution which mainly addresses to three phases viz., secure network access, key management and secure communication. The initial Handshake is done

using symmetric key which is the pre-shared key configured in devices a priory. Key management is done using polynomial scheme that guarantees sharing of secret bivariate by the domain manager of the network. Here, keys serve as root key material in MIKEY derivation. Whereas secure communication is guaranteed by DTLS record.

3.3 Message Authentication in CoAP

Research has been extensive to provide security using various means to the payload of the message but, securing the meta-data of the message is also of equal importance. Nguyen et al. [11] proposed a message authentication framework for CoAP message as there is an issue regarding the privacy of Meta information even though payload of message is secured. Protecting only the payload or certain data format still leaves a trail for an attacker to manipulate meta-data, which is a crucial part of CoAP message. Distinction between header parts of CoAP is needed in order to differentiate meta-data from payload which isn't present, the proposed research provides distinction between CoAP start header and CoAP header.

Considering an MITM model, an attacker can intrude due to known DTLS vulnerabilities. It is a complimentary to Transport layer security. The REST-ful CoAP message authentication protects and ensures authenticity by implying following steps:

1. Defines various message parts that are needed to be uniquely defined.
2. Implements REST-ful CoAP message signature generation algorithm.
3. Implements REST-ful CoAP message signature validation algorithm.

4 Research Issues in CoAP

CoAP being a standardized protocol for constrained devices, these is an extensive amount of research going on regarding various improvisations that can be made in order to hyper its reliability and efficiency in Internet of Things. Despite there is huge research completed and still going on, CoAP security still requires a lot more mining done to address some issues that are not taken care of. The most important drawback found in CoAP is that it lacks its own built-in security module, hence there is a necessity of bind some external security protocol or technology to obtain security in CoAP. DTLS is being stated as standard security solution for CoAP, but use of DTLS as security scheme also restricts us from leveraging all features of CoAP. There are still some unaddressed issues that remain as an open research challenge regarding Constrained Application Protocol.

– CoAP still posses high energy consumption, data loss and delay as DTLS posses heavy packet size.

Table 1. Study of existing security scheme for CoAP

Sr. no.	Research article topic	Year	Security scheme	Key management	Authentication mechanism	Message security	End-to-end security	Header compression	Protection from attacks
1	6LoWPAN compressed DTLS for CoAP [15]	2012	-	-	-	-	-	Yes	-
2	Securing IP based IoT With HIP and DTLS [4]	2013	Host Identity Protocol	AMIKEY	Yes	Yes	Yes	No	Yes
3	LITHE [14]	2013	DTLS	Yes	Yes	Yes	Yes	Yes	Yes
4	Secure multicast transmission [7]	2013	Batch Signature Verfiy	Public Key	Yes	Yes	Yes	-	Yes
5	Securing communication in 6LoWPAN with compressed IPSec	2013	IPSec	No	Yes	Yes	Yes	-	Yes
6	AuthLite [19]	2014	Symmetric Key	YES	Yes	Yes	Yes	Optional	No
7	Lightweight secure communication for CoAP enabled IoT using delegated DTLS [12]	2014	DTLS	Yes	Yes	Yes	Yes	No	Yes
8	Lightweight DTLS In CoAP based IoT [10]	2014	Tiny DTLS	Yes	Yes	Yes	-	No	-
9	Security analysis of DTLS structure and its application to secure multicast communication [16]	2014	Centralized Control Secure multicast Scheme	Yes	Yes	Yes	Yes	No	-
10	A decentralized approach for security and privacy challenges in IoT [18]	2014	Public Key Cryptography	Yes	Yes	YEs	-	No	-
11	REST-ful CoAP message authentication	2015	REST Signature [11]	YES	Yes	Yes	Yes	No	-
12	Lightweight security scheme for IoT applications using CoAP [20]	2014	Symmetric Key Based	Yes	Yes	Yes	Yes	New Header options	Yes
13	LESS - lightweight establishment of secure session	2015	Payload embedded response scheme [2]	Yes	Yes	Yes	Yes	-	Yes
14	A distributed security for resource constrained IoT devices [9]	2016	Transport Layer Security/Symmetric Encryption	Yes	Yes	Yes	Yes	No	-

- Being request/response protocol implies four round trips for initial authentication.
- DTLS defines to use Elliptical curve cryptography for key management but, there is requirement of a second thought over ECC technique as its practicality is questionable.
- A prime feature of CoAP, multicast messaging cannot be performed using DTLS and proves to be essential in IoT environments.
- DTLS lacks the support for group key management.

Although there are certain research proposals aiming towards alternative approach regarding CoAP security other than DTLS, those are mostly dependent of key management. In a network consisting of multiple nodes, distribution and management of encryption keys still persist as an important issue that awaits a reliable and efficient solution.

The wide range of security schemes mentioned above lack firm results on their resistance to various probable attacks on the network. We lack the knowledge of reliability of network and its security over a real-time network and traffic as results presented by researchers and authors are based upon lab experiments and simulation software.

No firm simulation evaluation criteria /frameworks are available to perform standardized output and signify the results. The implementation of IoT is mainly carried out using traditional networks i.e. connecting nodes and maintaining a server that records the behavior and performs necessary actions, as the technology of cloud is growing reliable and accessible easily, it is necessary to implement IoT over cloud to provide global access.

With an objective of performing an extensive survey with respect to various security mechanism for CoAP, a study was performed with its outcomes mentioned in Table 1. Depending on various mentioned parameters that are necessary or posses importance, this study examines various important security schemes back from 2012. This study help to understand the importance of various security parameters and advancements in security techniques.

5 Conclusion

Through this paper we surveyed and studied different techniques that are associated with Constrained Application Protocol to guarantee secure communication in Internet of Things. We measured out that DTLS is mentioned as a standard mechanism for securing CoAP protocol and it also provides the necessary security to some extent. But there are still some modifications required to reduce the cost of this heavy protocol with respect to the heavy handshake mechanism and packet size. We also came across various other security schemes that are lightweight but not yet standardized. The message authentication scheme studied provides protection to meta-data as well, which is a add-on in improving security in CoAP. Here, we also state various issue that still persist and need to be addressed to provide overall security to CoAP over IoT. Some techniques mentioned here are evaluated and verified to provide efficient results and reliability in

securing CoAP. We expect that this survey provide some valuable contribution and proper insights by documenting a very dynamic area of research in this era. This will definitely be helpful to the researchers to evolve with new solutions in the aspect of securing the IoT.

References

1. Arkko, J., Keränen, A.: CoAP security architecture (2011)
2. Bhattacharyya, A., Bose, T., Bandyopadhyay, S., Ukil, A., Pal, A.: Less: lightweight establishment of secure session: a cross-layer approach using CoAP and DTLS-PSK channel encryption. In: 2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 682–687. IEEE (2015)
3. Bormann, C., Hartke, K., Shelby, Z.: The Constrained Application Protocol (CoAP). RFC 7252, June 2014
4. Garcia-Morchon, O., Keoh, S.L., Kumar, S., Moreno-Sanchez, P., Vidal-Meca, F., Ziegeldorf, J.H.: Securing the IP-based internet of things with HIP and DTLS. In: Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks, pp. 119–124. ACM (2013)
5. Granjal, J., Monteiro, E., Silva, J.S.: Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun. Surv. Tutor. **17**(3), 1294–1312 (2015)
6. Ishaq, I., Hoebeke, J., Van den Abeele, F., Moerman, I., Demeester, P.: Group communication in constrained environments using CoAP-based entities. In: 2013 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), pp. 345–350. IEEE (2013)
7. Salem Jeyaseelan, W.R., Hariharan, S.: Secure multicast transmission. In: 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), pp. 1–4. IEEE (2013)
8. Karagiannis, V., Chatzimisios, P., Vazquez-Gallego, F., Alonso-Zarate, J.: A survey on application layer protocols for the internet of things. Trans. IoT Cloud Comput. **3**(1), 11–17 (2015)
9. King, J., Awad, A.I.: A distributed security mechanism for resource-constrained IoT devices. Informatica **40**(1), 133 (2016)
10. Lakkundi, V., Singh, K.: Lightweight DTLS implementation in CoAP-based internet of things. In: 2014 20th Annual International Conference on Advanced Computing and Communications (ADCOM), pp. 7–11. IEEE (2014)
11. Nguyen, H.V., Iacono, L.L.: REST-ful CoAP message authentication. In: 2015 International Workshop on Secure Internet of Things (SIoT), pp. 35–43. IEEE (2015)
12. Park, J., Kang, N.: Lightweight secure communication for CoAP-enabled internet of things using delegated DTLS handshake. In: 2014 International Conference on Information and Communication Technology Convergence (ICTC), pp. 28–33. IEEE (2014)
13. Rahman, R.A., Shah, B.: Security analysis of IoT protocols: a focus in CoAP. In: 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC), pp. 1–7. IEEE (2016)
14. Raza, S., Shafagh, H., Hewage, K., Hummen, R., Voigt, T.: Lithe: lightweight secure CoAP for the internet of things. IEEE Sens. J. **13**(10), 3711–3720 (2013)

15. Raza, S., Trabalza, D., Voigt,, T.: 6LowPAN compressed DTLS for CoAP. In: 2012 IEEE 8th International Conference on Distributed Computing in Sensor Systems (DCOSS), pp. 287–289. IEEE (2012)
16. Shaheen, S.H., Yousaf, M.: Security analysis of DTLS structure and its application to secure multicast communication. In: 2014 12th International Conference on Frontiers of Information Technology (FIT), pp. 165–169. IEEE (2014)
17. Sheng, Z., Yang, S., Yifan, Y., Vasilakos, A., Mccann, J., Leung, K.: A survey on the IETF protocol suite for the internet of things: standards, challenges, and opportunities. IEEE Wirel. Commun. **20**(6), 91–98 (2013)
18. Skarmeta, A.F., Hernandez-Ramos, J.L., Moreno, M.V.: A decentralized approach for security and privacy challenges in the internet of things. In: 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 67–72. IEEE (2014)
19. Ukil, A., Bandyopadhyay, S., Bhattacharyya, A., Pal, A., Bose, T.: Auth-lite: lightweight M2Mauthentication reinforcing DTLS for CoAp. In: 2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops) pp. 215–219. IEEE (2014)
20. Ukil, A., Bandyopadhyay, S., Bhattacharyya, A., Pal, A., Bose, T.: Lightweight security scheme for IoT applications using CoAP. Int. J. Pervasive Comput. Commun. **10**(4), 372–392 (2014)