Security Scheme for CoAP in IoT : A Survey

Amit Mali
Department of Computer Engineering
Sardar Patel Institute of Technology
Andheri(W), Mumbai
Email: aamali3031@gmail.com

Dr.Anant Nimkar
Associate Professor
Department of Computer Engineering
Sardar Patel Institute of Technology
Andheri(W), Mumbai
Email: danantvnimkar@gmail.com

Abstract—**Internet of things is the fast developing network between smart products or smart things connected to the internet. It not only connects objects and people but also billions of gadgets and smart devices. IP-based communication protocols play a key role in enabling the ubiquitous connectivity of devices in the context of IoT applications. Rapid growth in IoT increases security vulnerabilities of the linked objects. For communication at the application layer, resource-constrained devices are expected to employ the Constrained Application Protocol (CoAP) that is currently being standardized at the Internet Engineering Task Force(IETF). 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 in-order 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.**

I. Introduction

The basic idea behind the Internet of Things (IoT) is the integration of all kinds of electronic devices into the Internet with the 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.TCP performance is known to be inefficient in wireless networks, due to its congestion control algorithm, and the situation is exacerbated with the low-power radios and lossy links found in sensor networks. Therefore, the connection-less UDP is mostly used in the IoT. One approach which fulfills these requirements is the architectural style 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 the aim to build a worldwide distributed system of interconnected physical objects.

The security of IoT is a vital topic, due to the fact that it deals with sensitive data that flows over the Internet.A study states that 70 percent of the ordinarily used IoT devices face security vulnerabilities such as insufficient authorization, lack of encryption, and insecure web interfaces.In other words, security, privacy and trust are the main elements that businesses have to focus on when implementing IoT. However, the biggest challenge is performance and speed, if security is applied. The IoT devices are light and therefore manufactured keeping in mind low processing power and higher memory capabilities so that users can communicate with each other to minimize delay, and without effecting overall throughput in a high packet loss environment

The Constrained Application Protocol (CoAP) is under standardization as an application layer protocol for the IoT.The IETF Constrained Application Protocol (CoAP) is an application layer protocol tailored to resource constrained devices and M2M applications. It allows communication over the Internet among IoT objects that support UDP and 6lowPAN, achieving low overhead and supporting multicast.CoAP architecture is divided into two layers: the lower message layer and the upper request/response layer. The message layer provides reliability and sequencing by means of a stop and wait protocol using the following types of messages: 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.

This article analyzes study from various available literature that are present and is, as far as our knowledge goes, the first survey focusing on security techniques for CoAP in the IoT. Other surveys do exist that, rather than analyzing the technologies currently being designed to enable Internet communications with sensing and actuating devices, focus on the identification of security requirements and on the discussion of approaches to the design of new security mechanisms [Reference on survey], or on the other end discuss the legal aspects surrounding the impact of the IoT on the security and privacy of its users.

Our article is organized as follows, Section II focuses on Security requirement for CoAP. In Section III we discuss various existing security schemes implied to CoAP and Section IV we enlighten some research issues still present in CoAP and we conclude the article on this in Section V.

II. Constarined Application Protocol

The application layer CoAP protocol was originally developed for web transfer with constrained nodes and networks in the Internet of Things. The protocol is a HTTP remarkable version to match the IoT requirement for low overhead and multi-cast support. CoAP depends on REST, a principle adopted from HTTP and embedded in UDP for the transaction. The initial reason for developing this protocol is to match the high requirement of IoT and the need for low rate and light protocol . Overall, the main features of CoAP are: it supports M2M requirements in constrained environments, UDP binding with optional supporting uni-cast and multi-cast requests, asynchronous message exchanges, low header overhead and parsing complexity, supports URI (Universal Resource Identifier) and content-type, and that it has simple proxy and caching capabilities.

A. *CoAP Architecture*

CoAP architecture split into two layers, message layer and request/response layer. The first layer is responsible for controlling the message exchange over UDP between two end points. The format of the message will be outlined next. While the second layer carries the request and response which hold the method code and response code in order to avoid issues such as the arrival of messages that are out of order, lost or duplicated. Thus, CoAP is a reliable mechanism with rich features such as, simple stop-and wait re-transmissions, duplicate detection and multicast support. CoAP uses a short fixed-length binary header and components, and messages are encoded in binary simple format . Figures show each component and the message format. The methods supported in CoAP are based on the REST-ful structure which are listed as follows:

- GET: operation to retrieve representation in resource identified by the URI request;
- POST: request the server to create a new subordinate resource under the requested parent URI;
- PUT: requests resource identified by the request URI to update/created with the enclosed message body;
- DELETE: requests resource identified by the request URI to delete.

B. *Security in CoAP*

Security for CoAP protocol requires the existence of an additional encrypted protocol as a cover, similar to traditional XML-data representations and protocols (e.g., HTTP and Transport Layer Security (TLS)). The CoAP uses a UDP protocol and encryption is most commonly accomplished using Datagram Transport Layer Security (DTLS) and sometimes with IPSec. A biding of DTLS to CoAP is required to secure the messages DTLS was initially designed for traditional networks. Therefore, porting the protocol as it is over resource constrained devices produces a heavyweight solution. DTLS headers are also too long to fit in a single IEEE 802.15.4 maximum transmission unit (MTU). Computation overhead of their DTLS handshake introduces a high energy consumption due to the use of RSA-based cryptography. DTLS is applied in the transport layer and the fundamental AES/CCM provides confidentiality, integrity, authentication, and non-repudiation. DTLS is based on the Transport Layer Security (TLS) protocol and not the application layer. DTLS records are 8 bytes longer than in TLS. Once a handshake on DTLS is completed an additional 13 bytes will be over headed overhead per datagram. The process of handshake comes in outgoing and incoming messages scenarios. In the incoming scenarios the protocol will verify through decryption and decompressing. For outgoing, the protocol applies encryption algorithm, add authentication code (MAC) and compress the message as shown in Fig. the header of 6LoWPAN compression employs 10 bytes and CoAP header 4 bytes, therefore the focus is to compress the DTLS .

The security in CoAP is still under discussion, even though DTLS is combined as a protection layer. The debate is the heavy cost of computation and high handshake in the message which causes message fragmentation. Many studies have proposed a solution to compress the DTLS which is covered in following section.Furthermore, key management is another drawback of the CoAP security which is a common issue in almost all Protocols.Raza et al. have proposed to adopt 6LoWPAN header compression for DTLS. They have linked compressed DTLS with the 6LoWPAN standard, achieving a 62 percent reduction in the number of additional security bits. Research is also carried out in order to introduce a low overhead security mechanism using symmetric key-based authentication and confidentiality for CoAP suitable for constrained sensor devices. Here symmetric key is used with Advanced Encryption Standard (AES)128 Cipher Block Chaining (CBC) mode. Having a different perspective Authors in HIP PAPER 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.

III. Existing Security Scheme for CoAP

This sections 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 attain 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.

A. ***Security Using DTLS***

CoAP proposes to use Datagram Transport Layer Security (DTLS) as a security protocol for automatic key management and data encryption and authentication. CoAP with DTLS support is termed as secure-CoAP (CoAPs). DTLS was designed for the Internet and not for resource constrained IoT devices. etc. that support DTLS. The Maximum Transmission Unit (MTU) for 802.15.4 is 127 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 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 save us upto 62 percent bits.

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 inorder to implement it as security support for CoAP. It is a very fruitful in all aspects being a novel method for secure CoAP over the Internet of Things. Evalutaion results of this technique over simulation environment in Contiki OS proved to device some fruitful results that showcased very less amount of bytes transferred resulting in an efficient and CoAP implementation. The header compression reduce a huge amount of traffic in the network, leading to a minimal energy consumption.In comparison to plain CoAP, the response time was drastically reduced which proves DTLS compresseion efficient in terms of energy comsumption. Also Lithe avoids fragmentation which results in fragmentation attacks over an IoT system. Figures depict the efficiency of CoAPs with compressed DTLS over plain CoAPs. **** Figure of results

To implement Secure CoAP park et. al. proposed a technique to separate DTLS protocol into handshake phase and encryption using a Secure Service manager (SSM).This results to overcome various problems in LLN such as data loss, delay that contribute in increasing the overhead in the network. This separation also prevents the system from DoS attacks as encryption phase is separated at host location. Proposed system is unlimited as to the choice of cipher suite. 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. The SSM and the constrained device are physically separate but can virtually be regarded as one system in a trusted relation via a pre-shared key.

B. *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 authentication with integrated key management. Exchanged symmetric key is used with Advanced Encryption Standard (AES)128 Cipher Block Chaining (CBC) mode. This method is payload embedded thus minimizing the handshaking overhead. It consists of following phases: secret distribution; session initiation; server challenge; sensor response.

Furthermore Bandopadhy et. al. came up with AuthLite[ ] that enables security to CoAP by providing object security. Security is provided by a symmetric key based security mechanism where key management is integrated with authentication. Along with mutual authentication-key exchange protocol a new header options in CoAP to minimize resource consumption. AuthLite is manned to protect the system threats of DoS attack, replay attack, meet-in-the-middle attack, and information disclosure attack. When evaluated against DTLS based CoAP system it is observed that AuthLite has higher performance and less losses in a pre-shared key mode. AuthLite combined with DTLS provides a perfect security solution that provides mutual authentication layer and also protect from various attacks.

Obtaining allround security in constrained environment is challenging because existing IP security protocols do not offer all required functionalities and typical Internet solutions do not lead to the best performance. To overcome this situation a new variant of host Identity Protocol based on Pre-Shared Key(PSK) is stated which introduces a cryptographic namespaces of stable host identifiers between network and transport layer to improve the performance and reliability of constrained networks. Oscar et. al. 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 i.e. 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 layer

C. *Message Authentication in CoAP*

Nguyen et. al. proposed a message authentication framework for CoAP message as there is an issue regarding the privacy of Meta information even though payload of a 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 isnt 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 [ ]. The REST-ful CoAP message authentication protects the integrity 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.

It is a complimentary to Transport layer security

IV. 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 the previously analyzed research proposals, various issues remain to be addressed in the context of CoAP security 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. DTLS being a request/response protocol implies four round trips for initial authentication.
- DTLS defines to use Elliptical curve cryptography for key management but, The support of ECC public-key cryptographic on 6LoWPAN environments requires further investigation, as the viability of ECC cryptography on constrained sensing platforms is not currently consensual.
- The employment of DTLS is not well suited to the usage of CoAP proxies in forward or reverse modes.
- 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 towards CoAP security other 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.

V. Conclusion

Through this paper we surveyed and studied different techniques that are associated with CoAP 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] kns,"lsk n", *p;k's'dfk,asz*,kdojf,2144.