ISSN No. 0976-5697

Volume 8, No. 5, May – June 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Security in Application Layer Protocols for IOT: A Focus on COAP

Makkad Asim
Department of Computer Science and Engineering
Institute of Technology, Nirma University
Ahmedabad, India - 382481

Abstract: Security is important at everywhere like software, hardware,Internet of Things (IoT) or Web of Things (WoT) is a wireless network between smart products or smart things connected to the Internet. IoT is emerging technology. IoT can be classified into software components and hardware components. The focus of the proposed project is on CoAP (Constraint Application Protocol) protocol which falls under software components. CoAP is web-based protocol which provides Datagram Transport Layer Security (DTLS) security. DTLS only provide security in unicast message, because DTLS do not support multicast. To provide security in multicast messages, the proposed solution is to distribute session keys using key distribution center. Using provided session keys user encryption or decryption multicast messages.

Keyword: Internet of Thing (IoT), Application layer protocol, CoAP, DTLS, Security.

I. INTRODUCTION

Security is main problem in every communication. There are many different type of attacks like Man-in-middle attack, eavesdropping, data modification, application layer attacks, sniffer attacks, IP spoofing, password based attack and denial-of-service attack use to interrupt communication. The IoT is based on a wide range of semiconductor advancements, including power administration gadgets, sensors and microchips. IoT is required to offer propelled availability of gadgets, frameworks and administrations that go past machine-to-machine (M2M) correspondences and spreads an assortment of protocols, domains and applications. IoT contains low power gadgets and IPv6 availability amongst every single gadget.

Figure 1 shows IoT protocol stack that include application layer protocol (like CoAP, MQTT, XMPP, AMQP), Transport layer protocol (like UDP, DTLS), Internet layer Protocol (like RPL, 6LoWPAN) and Network/Link layer Protocol (like IEEE 802.15 Series and IEEE 802.11 Series) [1]. This research is more inclined toward web-based protocols that is application layer protocol. In application layer protocol work is being done on CoAP and its security DTLS. The CoAP protocol is used for application like health-care, parking system and home management etc.

Layer	Protocols
Application Layer	CoAP, MQTT, XMPP, AMQP, RESTFUL, Websockets
Transport Layer	UDP, DTLS
Internet Layer	RPL, 6LoWPAN
Physical/Link Layer	IEEE 802.15 Series, IEEE 802 11 series

Figure 1: IoT protocols stack[1]

Definitions - IoT is the inter-networking of physical devices, vehicles, buildings, and other items-embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data [3]. In short IoT connect with each and every object and communicate with each other anywhere and anytime, so IoT do just like M2M communication. In other word Internet that make things or Objects are smart call IoT. Security is one of the need of every application or software. Cryptography algorithm (encryption and decryption algorithm) provide network security or provide security in communication called end to end security. Every application or software have different algorithms. IoT need low constrain, complexity and power algorithms.

II. RELATED WORK

IoT is emerging technology.. By studying paper regarding IoT and IoT protocol related IETF standards paper show application layer protocols focus basically on message exchange between applications and the internet, summary of application layer protocol show in Figure 2. In application layer protocols, for this research we chose CoAP for communication and for security of CoAP used DTLS transport layer protocols [2] [5].

Protocols	Transport	QoS	Architecture	Security
CoAP	UDP	YES	Request/Response	DTLS
MQTT	TCP	YES	Publish/Subscribe	TLS/SSL
XMPP	TCP	NO	Request/Response Publish/Subscribe	TLS/SSL
RESTFUL	HTTP	NO	Request/Response	HTTPS
AMQP	TCP	YES	Publish/Subscribe	TLS/SSL
Web socket	TCP	NO	Client/Server Publish/Subscribe	TLS/SSL
DDS	TCP/UDP	YES	Publish/Subscribe	TLS/SSL
SMQTT	TCP	YES	Publish/Subscribe	It have own

Figure 2: Summary of application layer protocols [1]

As CoAP is a RESTFUL (Representational State Transfer) web exchange convention for use with compelled systems. CoAP utilizes request/response model of approach same as HTTP show Figure 2. It is intended for obliged systems with low overhead and lower impression. A few focuses for CoAP that improves convention contrasted with HTTP are CoAP runs over UDP (User Datagram Protocol) that helps to avoid costly TCP handshake before data transmission, A very efficient RESTFUL protocol, Asynchronous transaction model, Easy to proxy to/from HTTP, URL support, Security binding to DTLS, CoAP has minimal header format that saves IoT of power for constrained nodes compared to running HTTP in that constraint nodes, Generally CoAP used DTLS with PSK for security and Support reliability and multicast [4].

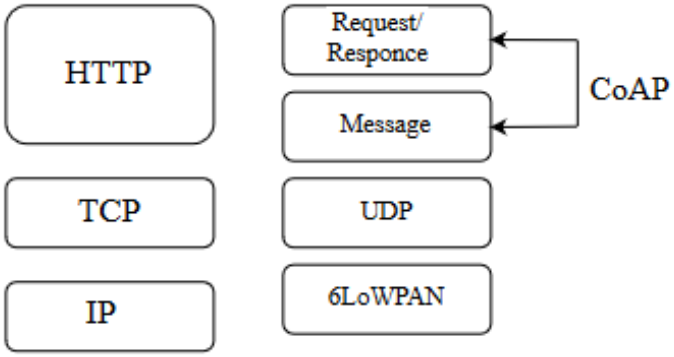

Figure 2: HTTP and CoAP protocol stack

In CoAP have many done work here some of the work are describe like Scalable Cloud Service with CoAP, NAT Issues, Web based Monitoring in Health care and End-to-End Security,

In web based monitoring in Health care, Health-care application like heart rate, ECG, blood pressure, level of glucose or oxygen, all this application problem solved by CoAP.Those applications have many problem regarding communication between sensor and server [11] [12]. To those problem overcome with CoAP. CoAP used to remote health-care monitoring system that provides the patient's condition through web browser. There are sensor collect data of patient and transferred to various IP end-devices. Those are communication use 6LoWPAN network to communicate with server. Study was done on CoAP based communication papers. Most of paper have one common problem is the security. Secure data, secure end-to-end communication, hardware security and many problem regarding to security.

In end-to-end security overcome those problem regarding security in CoAP. Surveying paper DTLS only provide unicast communication security. In this paper DTLS used LESS algorithm which protect the session key during exchange, the outgoing messages are forward to DTLS which then forewords them to the destination in protected mode. The incoming CoAP messages will protect by DTLS layer at first then it will be headed to CoAP layer [10]. LESS algorithm is nothing but non-blocking algorithm. This method is existing scenario of unicast message security in end-to-end communication.

III. PROBLEM STATEMENT

Security is the main issue in everywhere and needs of every application or software and hardware. There are the different type of security available for defending thread like third party authentication (the third party provide secure log-in like Google, Facebook) and use some anti-virus or anti-malware software. In this research, secure communication between one to many devices in IoT. For securing communication author already secure unicast CoAP message for communication between two devices. Secure unicast message author uses LESS algorithm (or non-blocking algorithm) as per show previews chapter how author secure single client-server communication [10]. Show Figure 3, for how unicast communication done and Figure 4, for how DTLS provide security in between two nodes.

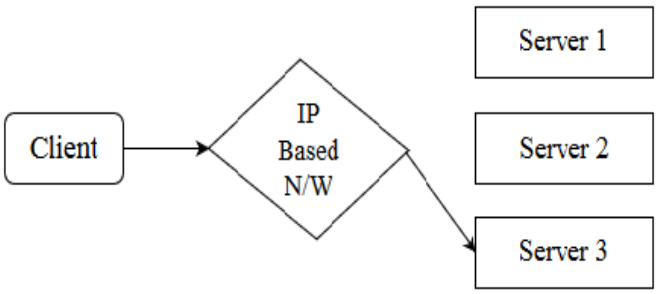

Figure 3: Unicast communication

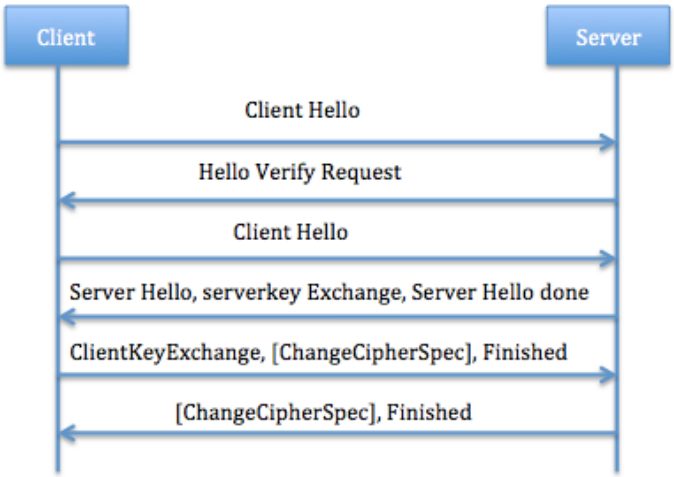

Figure 4: DTLS Handshake [10]

By studying all survey paper and existing work show in Figure 3 and Figure 4, DTLS (Datagram Transport Layer Security) protocol only secure unicast communication (client-server communication). DTLS done not provide security in multicast (one to many communication) because CoAP support multicast communication but DTLS does not support multicast communication.

So problem statement is "By studying all survey paper conclude that, DTLS only secure unicast message(or single client-server communication). Which is not provide multicast communication service because DTLS do not

support multicast. This research provide secure multicast service in IoT."

IV. PROPOSED SOLUTION

Multicast communication is generally one to many communication and one of the best examples of multicast communication is multimedia [8] [9]. The CoAP is similar to client/server HTTP model. In multicasting, the client can send multiple requests to the server but in IoT, CoAP is request/response model so, here client node send multiple requests to server node and also server response to the client. To understand all this scenario show Figure 5.

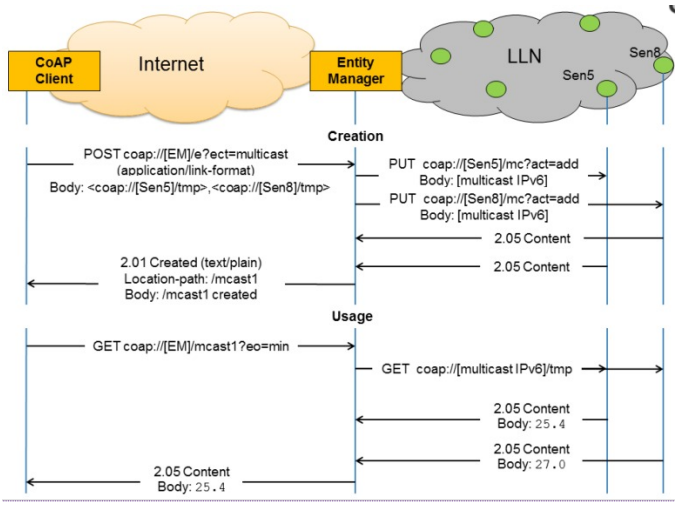

Figure 5: Multicast communication in IoT [8]

To secure multicast communication we have to generate session keys for every node. In this section, define the algorithm to secure multicast communication in IoT. For securing communication make KDC (Key Distributed Center) distribute session keys to every nodes or sensor. After generating session keys for every node do encryption for each note and create encrypted messages. The same method at decryption side using same KDC algorithm generate session keys then decryption with the encrypted message.

IoT needs low complexity, network capability and standardization to implement any IoT-based algorithm and equipment. That's why design algorithm contains low complexity. In IoT communication work under CoAP protocol and CoAP provide DTLS protocol based security. For multicast communication, DTLS do not support multicast, so here used TLS/SSL protocol based security for securing multicast communication. Secure multicast communication implementation and analysis describe in our next paper come in same journal in next issue. Here we implement secure unicast communication using LESS algorithm.

V. IMPLIMENTATION

For implement or test LESS algorithm to secure unicast communication we used contikios with cooja simulator as a Implementation tool.

Contiki is open source operation system. It is lightweight OS and implement in C programming language. It is purposely developed and created for the low-power devices in constrained environment [6]. It is connects constraint node to Internet. It provide powerful low power Internet communication. It supports IPv6 and IPv4 standard along with 6LoWPAN, CoAP and RPL. It can be used in commercial, non-business and full source code is simple accessible [7]. Contiki application are written is C programming language and used cooja simulator for simulation purpose so that network can be emulated before burned into hardware.

Cooja is network simulator for Contiki which allowed larger and small network for simulation. Cooja control and analyze contiki system via few function. In front-end interface, cooja used combination of java codes [7]. It is a cross-level simulator, built in Contiki OS. It Provide the simulation on network level, OS level and machine code level. It is network simulator for contiki which allowed big and tiny network for simulation.

First we create unicast communication in cooja simulator show Figure 6 to topology of unicast communication. In this topology one is client and another one is server and green circle is radio environment for strong communication.

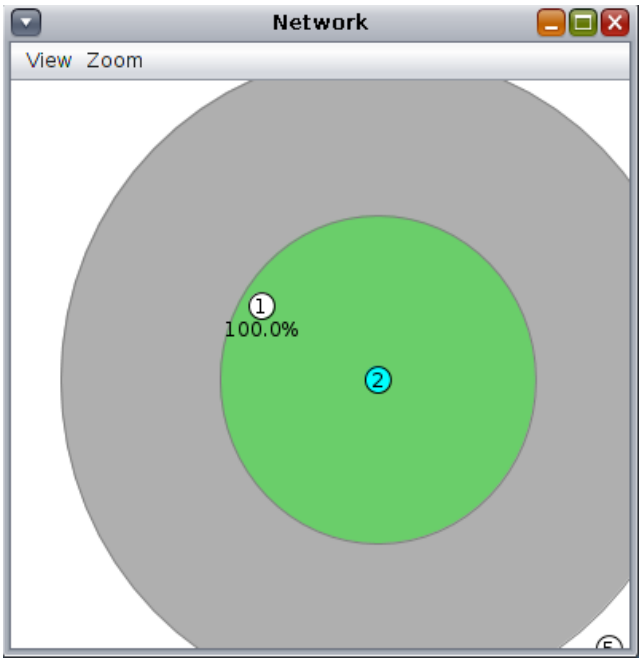

Figure 6: Unicast topology in cooja

Now apply LESS algorithm to client and server both side and below Table 1 show implementation result. In cooja simulator take 850 ms time for session key generation.

Table 1: Encryption and Decryption time for node

Node	Encryption time (ms)	Decryption time (ms)
1	1008	1125

Now we analyze encryption and decryption time for different file size show in Table 2.

File size (bytes)	Encryption time (ms)	Decryption time (ms)
10	1008	1125
20	1098	1204
30	1200	1296
40	1283	1379
50	1390	1459
60	1471	1575
70	1542	1655
80	1612	1730
90	1698	1810
100	1785	1894

Table 2: Encryption and Decryption time in different file size

This is how we implement or test algorithm in cooja simulator and analyze performance of algorithm.

VI. CONCLUSION & FUTURE WORK

There are three major components for implementing IoT on different applications: Security, Privacy and Trust. While increasing the growth of IoT, security is more important for reliable data transferred among the billions of smart objects. In these research, we concentrate on CoAP protocol application layer protocol on IoT devices. Having light weight and consume low energy, CoAP is used on many applications of IoT. To secure data transferred, CoAP combined with DTLS protocol named as Datagram Transport Layer Security protocol as the security agent. The heavy weight of DTLS protocol used to protect the communication between smart objects on IoT.

On some of the area, DTLS may not secure for reliable data transferred and can be considered as the threat for the protocol. DTLS do not supporting for multicast messages in communications on IoT. Having a lack of security in DTLS protocol, Random session key generation stream algorithm is used to protect the communication among the object's security. The various implementation of CoAP protocols on IoT may lead towards the secure communications among smart objects.

In future work we implement multicast communication then design cryptography algorithm then test or implement algorithm in cooja simulator and analyze result.

I. REFERENCES

[1] Makkad Asim “A Survey on Application Layer Protocolsfor Internet of Things (IoT)” in International Journal of Advance Research in Computer Science. March- April 2017 Volume 8. No.3ISSN No. 0976-5697.

[2] S. Kraijak and P. Tuwanut. “A survey on IoT architectures, protocols, applications, security, privacy, real-world implementation and future trends”. In: 11th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2015). Sept. 2015, pp. 1–6. DOI: 10.1049/cp.2015.0714.

[3] Administration. Internet of Things. 2016 (accessed November 3, 2016).
URL: https://en.wikipedia.org/wiki/Internetofthings

[4] Xi Chen. “Constrained Application Protocol for Internet of Things”. URL: https://www.cse.wustl.edu/~jain/cse574-14/ftp/coap/

[5] Stan Schneider.Understanding The Protocols Behind TheInternet Of Things.
URL: http://electronicdesign.com/iot/understanding-protocols-behind-internet-things

[6] Contiki tutorial – contiki, 4 November 2014.

[7] A. Sehgal, Using the contiki cooja simulator, 29 October 2013

[8] D. M. Mani, “Secure multicasting for wireless sensor networks,”International Journal of Computer Science and Network Security (IJCSNS), vol. 14, no. 11, p. 70,2014.

[9] M. I. D. P. Ishaq I, Hoebeke J, “Experimental evaluation of unicast and multicast CoAP group communication,”Sensors (Basel, Switzerland), 16.7 (2016).

[10] R. A. Rahman and B. Shah, “Security analysis of iot protocols: A focus in coap,”in2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC),pp. 1–7, IEEE, 2016.

[11] . A. Khattak, M. Ruta, E. D. Sciascio, and D. Sciascio, “Coap-based healthcaresensor networks: A survey,” inProceedings of 2014 11th International Bhurban Conference on Applied Sciences Technology (IBCAST) Islamabad, Pakistan, 14th - 18thJanuary, 2014, pp. 499–503, Jan 2014.

[12] D. Ugrenovic and G. Gardasevic, “Coap protocol for web-based monitoringin iot healthcare applications,” in2015 23rd Telecommunications Forum Telfor(TELFOR), pp. 79–82, Nov 2015.