Security-aware CoAP Application Layer Protocol for the Internet of Things using Elliptic-Curve Cryptography

Firas Albalas[1], Majd Al-Soud[1], Omar Almomani[2], and Ammar Almomani[3]
[1]Department of Computer Science, Jordan University of Science and Technology, Jordan
[2]Network Computer and Information Systems Department, the World Islamic Sciences and Education University, Jordan
[3]Department of Information Technology, Al-Balqa Applied University, Jordan

Abstract: *Currently, the concept of the Internet of Things (IoT) has become more noticeable where it is being used in all aspects of life, such as home automation, smart cities, military surveillance, security, agriculture, healthcare, etc., However, the heterogeneity of the constrained devices and the complexity of the internet bring up the need for a security system to secure all the communications, data and participating things. In this paper, This paper proposed a lightweight secure Constrained Application Protocol (CoAP) using Elliptic Curve Cryptography (ECC) to transport security between IoT objects and the Resource Directory (RD). The advantage of using ECC is its compact key size enabling it to utilize a smaller key size compared to the other identification methods such as Rivest-Shamir-Adleman (RSA). This work mainly has two parts; the first part implements the CoAP using ECC and using RSA algorithms where the results have proven that using ECC much better than RSA in terms of energy saving. The second part of this paper shows the proposed evaluation function and focuses on the security services that were applied in the proposed protocol. The results show that authentication achieved a 75.3% energy savings, data integrity had a 55.7% energy saving and confidentiality achieved a 47% energy saving.*

Keywords: *IoT, CoAP, ECC,energy saving, security, IoT.*

Received February 12, 2018; accepted April 18, 2018

1. Introduction

In 1999, Internet of Things (IoT) was at first originated by Kevin Ashton in the supply chain industrial management context [14]. Nevertheless, throughout the past years, this definition has been extended to cover a wider range of fields and applications such as transport applications, healthcare applications, utility applications, etc., In spite of "Things" definition has completely changed as new technologies evolved the trends of creating and making computer sense information without any type of human assistance or intervention remain the same.

The growth and evolution of the Internet to a network of interconnected things which can not only collect information from the around world through sensing as well as interacting with the physical environment by controlling, actuating, sending or receiving commands, but it can also use the available internet standards to supply services for many applications such as data transfer, information analytics, and communications. With the large spread of the devices that were enabled by the wireless networks, actuator nodes, and embedded sensors, IoT has emerged from its cradle as well as is on the brink of transforming the traditional Internet to a complete integrated interconnected future Internet [8].

The recent revolution of the Internet has resulted in new connections between groundbreaking scale and people. The next Internet revolution will guarantee connections between objects and things to establish smart environments. In 2011, the number of the interconnected objects and devices in the world started to exceed the people number [7].

Currently, there exist 9 billion connected devices to the internet and by 2020 it is expected to reach up to 24 billion devices. This neutralizes to $1.3 trillion total revenue opportunities for the operators of the mobile networks that transferring essential segments such as utilities, automotive, health and consumer electronics according to GSMA association [1].

IoT's has added a new dimension to the world of Information and communication technologies, by creating a new form of communication between things and people, between things themselves, connecting everyday devices such as smart-phones, internet TV's, sensors and actuators to the internet. To enable these low-power devices with limited processing capabilities to participate in the IoT, standardization organizations and the research community have defined several architectures and protocols. Therefore, the concept of 6LoWPAN originated, which stands for IPv6 over Low Power Wireless Personal Area Network, to enable IPv6

connectivity even to the smallest objects. IPv6 and IPv4 are used to deliver data for the local area, metropolitan area and wide area networks.

The 6LoWPAN protocol enables IPv6 packets to be sent to and received from over IEEE 802.15.4 based networks, which provides the devices with sensing communication-ability in the wireless domain. However, to cope with constrained resources and the size limitations of IEEE 802.15.4 based networks, the 6LoWPAN group has defined the header compression mechanisms, although the standard 6LoWPAN already defines the header compression format for the IP header, IP extension headers, and the UDP header. For example, the header compression mechanisms standardized in RFC6282 can be used to provide header compression of IPv6 packets over IEEE 802.15.4 based networks.

The reason for making the IoT's IP connected is the need of the applications for wireless internet connectivity at lower data rates for devices with limited constraints. Examples of such applications are smart cities, smart grids, home automation, e-healthcare, and others.

Since there are a lot of devices that are unable to communicate in an efficient way with constrained resources, the Internet Engineering Task Force (IETF) came up with a lightweight protocol CoAP. CoAP is considered as a replacement of HTTP protocol, to be used as the IoT application layer protocol. This protocol was designed specifically to meet the requirements of the constrained devices such as low overhead, simplicity, and multicast support. Furthermore, as there exist many constrained devices in buildings and vehicles, IPv6 is used for the IoT implementation, since it provides a larger address space to let more devices get connected to the internet. CoAP was also developed as a candidate protocol to connect energy–constrained devices to the internet. Table 1 [7] presents a comparison of the resource consumption between HTTP and CoAP.

Table 1. Resource consumption comparison between HTTP and CoAP [7].

Parameters	HTTP	CoAP
Bytes per transmission	1451	154
Power (mw)	1333	151
Lifetime (days)	0.744	84

Although IP networking brings new opportunities and improvements in our daily life, security remains a concern that has to be tackled. Past experiences have proven that designing a right security protocol is a difficult and error-prone process. Thus, when researchers were concerned about security in IoT, they were designing lightweight variants and porting them to constrained devices leading to a situation where security doesn't keep up, although standardized Constrained Application Protocol (CoAP) completely supports the requirements of the application.

Many research works have been done focusing on the security methods for IoT, which can face different security attacks that affects the functionalities and services provided by the IoT network. In this paper, different available security solutions have been analyzed for the communication between devices. Furthermore, a secure CoAP is proposed using Elliptic Curve Cryptograph (ECC). Then compared with CoAP using Rivest-Shamir-Adleman (RSA) algorithm.

The arrangement of this paper is as the following: Section II, views the literature review including background about CoAP, ECC algorithm, and RSA. Section III introduces the reader to the proposed algorithmic design and equations for the proposed secure CoAP. Section IV shows the results that compare the CoAP using RSA and the proposed secure CoAP using ECC. Section V, concludes this paper and shows the future work.

2. Literature Review

Designing a secure and trustworthy network, and providing end-to-end security during communication is probably the most challenging task in IoT. And in IoT, security becomes more important, since most of the exchanged information are in general sensitive and private. A lot of problems can occur by adding security to the network; the most important is the consumed power since it is directly related to the lifetime of the network. Recently, a lot of research has been done to investigate this issue, by trying to adapt the traditional security methods and techniques into IoT. In this section, brief background information about CoAP protocol is presented and discuss some approaches and solutions provided to secure the constrained environments.

3.1. Background

Integrating security into CoAP is a very difficult thing to do since IoT is a heterogeneous network [2]. Next subsections provide a brief introduction of CoAP, its operations and methods, message format and transaction model:

1. *Constrained Application Protocol (CoAP)*: recently, the IETF working group developed a new application layer protocol called CoAP [10], that aims to integrate the Representational State Transfer (REST) architecture into Low Power and Lossy Networks (LLNs). This protocol applies some features of HTTP protocol but extending its role to be employed in 6LowPAN taking into consideration different constraints such as energy and Machine to Machine (M2M) applications used in the IoT environment.

CoAP protocol is developed to work with the transport layer User Datagram Protocol (UDP) which is very

suitable for client-server communication through an acknowledged datagram [11].

The following features are implemented in CoAP protocol which makes this protocol suitable for constrained networks and furthermore suitable for networks created for industry purposes:

- It is simple and compact protocol to be fit for the limited features in constrained nodes including memory and number of bits in its microcontroller.
- The small packet header size which makes it suitable for limited bandwidth nodes with less packet loss.
- The ability to cache the most recent responses to be forwarded later to nodes came back to work after sleeping mode finished.
- The ability to send the large packet using UDP protocol with back-off time for lost and non-delivered packets.

As any other Application Layer protocols; CoAP implements a set of operations that are required to control and use the available resources. These operations are similar to HTTP operations because as mentioned before, CoAP is developed to accomplish the HTTP job for constrained nodes running over 6LowPAN networks.

2. *CoAP Messages Format*

Figure 1shows the CoAP header format including a set of fields and the number of bits for each.

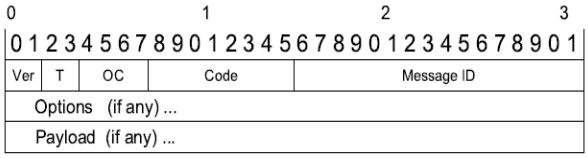

Figure 1. The format of CoAPMessage [6].

The following is a detailed description for CoAP header fields:

- *Ver (Version)*: the CoAP version and in this simulation the value is 1 and the remaining values are reserved for future versions.
- *T (Type)*: the type of CoAP messages. Whereas Confirmable (CON) message has 0 value, Non-Confirmable (NON) message has 1 value, Acknowledgment (ACK) message has 2 value, and Reset (RST) message has 3 value.
- *OC (Option Count)*: indicate whether the options field will be used or not.

3. *CoAP Transaction Model*: As mentioned before, CoAp has a client/server approach data transmission as shown in Figure 2.

The client asks for a service by sending a request and the server replies by a response message to serve the client. Also, this protocol supports an asynchronous transaction over the UDP. This is done by using the previously mentioned messages.

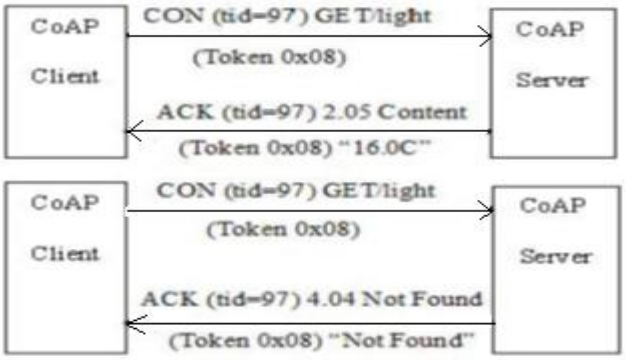

Figure 2. CoAP transaction model.

3.2. Related Work

Many researchers are currently focusing on the IoT security issue; therefore, a lot of security solutions have been provided. As proposed by Raza *et al.* [16], they proposed mechanisms to exploit the compression capabilities of 6LoWPAN to compress the DTLS (Datagram Transport Layer Security) headers and messages. Since DTLS was designed for the internet not for constrained IoT devices, it is a heavyweight protocol with too long headers to fit in a single IEEE 802.15.4 MTU (Maximum Transmission Unit). Therefore, 6LoWPAN is used in IoT to compress the long IP layer headers. The authors also defined the Record header, Handshake header, ClientHello message, and the ServerHell o message and their compression techniques. One of the results showed that for the DTLS record header the number of additional bits can be reduced by 62%. However, their study faces a weakness that it did not go further to ensure that compression would not affect the security.

As proposed by Kothmayr [12] and proposed by Kothmayr *et al.* [13], a security solution based on RSA, the most widely used public cryptography algorithm. Their goal was to achieve high interoperability and low overhead. However, because of the overhead of DTLS handshake process, RSA consumes a large amount of energy which considers a weakness feature for IoT devices.

Therefore, as proposed by Raza *et al.* [17], another solution by using DTLS compression as well. The authors proposed Lithe – an integration of DTLS and CoAP for the IoT. Lithe consists of four components: DTLS, CoAP, DTLS header compression (using 6LowPAN), and a CoAP-DTLS integration module which was developed to allow the application to access CoAP automatically. The evaluation results show significant gains in the processing time, network response time and energy consumption by reducing the packet size. With this work, they were able to avoid fragmentation or decrease the number of fragments by using compression when the payload was slightly above the fragmentation threshold.

As proposed by Brachmann *et al.* [5], a short outline to ensure a secure IP-based Internet of Things was

presented. The authors specifically discussed two issues that need to be solved to secure the communication links: end-to-end security and secure group communication. End-to-end security goal is to achieve a completely secure communication between an HTTP and CoAP entity using the Datagram Transport Layer Security-Pre-Shared Keys (DTLS-PSK) protocol, and the 6LoWPAN Border Router, which acts as a proxy. As for group communication security, the goal is to establish a secure connection using DTLS for a group of devices with a single session key. Different available approaches have been identified to achieve this goal, which needs to be analyzed according to the application requirements.

As proposed by Alghamdi *et al.* [3], the authors analyzed two of the known security protocols that can be used to secure CoAP networks: DTLS and Internet Protocol Security (IPSec). Their analyses were based on the X.805 standard which with its architecture can provide a complete top-down systematic method to correct, predict, and detect the security vulnerabilities of the network. They highlighted the advantages of the two security protocols and mentioned their drawbacks as well. And since these protocols were not designed to deal with constrained environments the authors mentioned the reasons why they were not the most optimized solutions for CoAP security and argued for the need of a new lightweight and secure version of CoAP, which addresses the issues of the above-mentioned protocols.

As proposed by Ukil *et al.* [18], the authors proposed a lightweight security approach using AES (Advanced Encryption Standard) 128-bit symmetric key algorithm. They came up with an Auth-Lite approach which enables the authentication mechanism, and by modifying the CoAP header, they came up with CoAP-Lite which enables lightweight security for CoAP. But this approach faces a weakness that it can only be used in vehicle tracking systems, and it depends on the application, so for other applications it may or may not be efficient.

As proposed by Bhattacharyya *et al.* [4], the authors proposed a different solution based on the idea of session security. They used the LESS (Lightweight Establishment of Secure Session) algorithm a cross-layer approach using CoAP and Datagram Transport Layer Security -Pre-Shared Keys (DTLS-PSK) channel encryption. In the path of communication, LESS protects the session key during message exchange, enables integrity during session establishment using Advanced Encryption Standard-Cipher Block Chaining Message Authentication Code (AES-CCM) rather than Advanced Encryption Standard-Cipher Block Chaining (AES-CBC) encryption, and enables separation in key used in reverse paths of communication. The algorithm consists of six steps: pre-sharing secret, session initiation, server challenge, client response and challenge, client authentication and server authentication. The results show that less outperforms DTLS-PSK in all aspects. First, the full session was performed in two request/response steps in LESS compared to six in DTLS. And since each step may consist of more than one message in DTLS, physically fragmented datagrams may be transmitted. On the other hand, LESS was implemented as small CoAP messages which guarantee no or little fragmentation. But this solution works only with unicast security, therefore multicast security is still an open challenge [15].

In this work, another security solution is proposed using ECC, which is an Asymmetric key based security mechanism based on some algebraic structure. The next section shows the details of the proposed solution.

3. Proposed Algorithmic Design

This section mainly describes the algorithmic design that is proposed in order to secure communications between the objects in IoT networks and from the other hand to reduce consumed energy while these objects are communicating with each other. As mentioned in the literature, there are many methods that secure communications in IoT networks but each has drawbacks. In order to overcome these limitations, the proposed approach uses Elliptic Curve Cryptography (ECC) method that depends on elliptic curves' algebraic structure over finite fields. The main usage of elliptic curves in this approach is generating private and public keys. Also, they are used in encrypting and decrypting the messages. This algorithm was used in this proposed solution to reduce the needed computational power since the key size in this algorithm has the smallest size comparing with other cryptographic algorithms. In this paper, we used the same Adoptive energy model that was proposed.

Algorithm 1: the proposed secure CoAP using ECC

In COAP protocol if a service agent (SA) node went to send a message to the resources directory (RD).

1) *SA sends request for public key (PU) of the RD*
   SA ⟶ *RD:* PU_{RD}
2) *RD generate public key PU and private key PR with help of EEC*
3) *RD sends public key to SA*
   SA ⟶ *SA:* PU_{RD}
4) *SA encrypts the message with the received RD public key* PU_{RD}
5) *SA sends encrypted message to RD*
   SA ⟶ *RD: CT*
6) *RD decrypts the message with its private key* PR_{RD}

3.1. ECC Algorithm Key Generation

In ECC, two keys are generated; the private key and the public key. Each Service Agent (SA) sender node will encrypt its message with the Resource Directory (RD) public key. Then the RD, in its turn, will decrypt the message with its private key.

Then a random number *R* is selected within a range (n). After that, the next Equation (1) is used to generate the public key:

$$PU = R * P \quad (1)$$

Where R is a random number between 1 to n-1 as well as *P* refers to the point on the curve that is generated using the next Equation 2. And *PU* refers to the public key and n refers to the private key.

$$y^2 = x^3 + ax + b \quad (2)$$

3.1.1. Encrypting Messages:

Suppose "s" is the message that is sending from the SA. Let "s" has the point S on the curve "*E*". *K* is a random point is selected from the range [1-(n-1)] then the ciphertext that will be sent is in Equation (3) and in Equation (4):

$$C_1 = K*P \quad (3)$$

$$C_2 = S + K*PU \quad (4)$$

3.1.2. Decrypting Messages:

$$S = C_2 - d*C_1 \quad (5)$$

S is the original message that was sent.

Logical proof- to get the original message back:

$$S = C_2 - d*C_1 \quad (6)$$

$$C_2 - d*C_1 = (S + K*PU) - d*(K*P) \quad (7)$$

Where

$$C_1 \text{ and } C_2 \text{ from } 3 \text{ and } 4$$

3.2. RSA Algorithm Key Generation

As discussed before, RSA is the most known and popular cryptography algorithm that uses the public key. Three main authors came up with this algorithm in 1976 namely: Ron(*R*) ivest, Adi (*S*) Hamir as well as Leonard (*A*) dleman. Generally, the next Table shows the properties that are employed in RSA.

Table 2. Properties that are employed in RSA.

P and q, prime numbers	Private
r = p.q	Public
Φ(r) = (p-1)(q-1)	Private
PK (encryption key)	Public
SK (decryption key)	Private
X (plaintext)	Private
Y(ciphertext)	Public

In RSA, key pair generation can be done through next steps:

- *P* and *q* prime numbers must be defined
- *r = p.q* where q and p can't have the same value because that would make it easy to obtain *p* from the square root of *r*.
- $\Phi(r) = (p-1)(q-1)$
- *PK*, the public key is chosen to be relatively prime with $\Phi(r)$.
- Private key generation using the next equation:

$SK.PK = 1 \pmod{\Phi(r)}$.

Hence, *SK* can be obtained from the next equation

$$SK = \frac{1 + m\ \Phi(r)}{PK} \quad (8)$$

The integer m, through which *SK* can be obtained.

3.2.1. Encrypting Messages:

In RSA encryption, Plaintext is structured into block x1, x2 ,… such that each block represents a value in the range from 0 to *r*-1. After that, each block *xi* is encrypted to block *yi* with this equation:

$$yi = x_i^{PK} \bmod r \quad (9)$$

3.2.2. Decrypting Messages:

Following the next equation, each ciphertext block *yi* is decrypted into block *xi*.

$$xi = y_i^{SK} \bmod r \quad (10)$$

3.3. The Security Services

ITU-T X.800 has defined the security service as a service which guarantees to protect systems and/or transmitting data in protocol layers using a suitable security. Furthermore, it capable to provide two or more security requirements as declared by CNSS. Hence, the security service should grantees security requirements in order to transmit data or protect the systems. Security services can be listed as follows:

1. Authentication.
2. Availability.
3. Data confidentiality.
4. Data integrity.
5. Non-repudiation.
6. Access control.

Using ECC, four security services can be provided authorization, authentication, Integrity and Confidentiality as security services to the IoT networks as follows:

1. *Confidentiality*: using this security service any unauthorized node is denied from accessing the data.
2. *Authorization*: this security service gives each node a unique key pair (public and private) in order to make encryption and decryption.
3. *Integrity*: assurance that messages received by a destination node have not been changed in transit either through collision or via a deliberate tampering by an untrusted node by using a hash chain and MAC list.

4. *Authentication*: this service is achieved by employing public key, if any malicious/ anonymous node needs to communicate with network nodes then it needs the public key pair of the authorized node.

4. The Evaluation Function

In this work, an evaluation function is proposed that mainly uses three input parameters:

1. *The message length*: in order to save more energy and as a result to extend the network lifetime, this parameter was used in this work evaluation function since encrypting the messages plays a major role in saving the network energy. Hence, message length consideration is essential in order to reduce energy consumption.
2. *The security service*: the main purpose of the communication is determined by this value. The communications have purposes and a needed security service is linked with that purpose. Nevertheless, to make testing easier, this paperwork has one security service for each communication. As mentioned before, at the initial step of this paper communications, the security service is selected and initialized. The service agent selects the security service and the evaluation function select the service algorithm in the proposed security-aware CoAP protocol.
3. *The last parameter is the residual energy*: this is used for the energy resides in the service agents with the purpose of reducing energy consumption and increase the network lifetime of IoT. Whenever the service agent has low energy amount, then the proposed evaluation function will choose the least energy security service.

The proposed evaluation function that used in this work can be formatted into the following equation:

$$F(x) = security\ service + \frac{Residual\ energy}{Message\ length} \qquad (11)$$

In Cooja, measuring the residual energy can't be done directly, for that reason the function Energest is used which is available in Contiki to measure the power consumption in the node. However, because of the use of TMote sky in the experiments; additional power consumption sources should be considered, which are: CPU power consumption, transmitting power consumption, receiving power consumption, and the power consumption during the sleep mode. In order to do that, the following calculations are used:

1. The CPU power consumption value equals 1.8 multiplies by the energy estimation consumed in the CPU.
2. Sleep mode power consumption value equals 0.0545 multiplies by the energy estimation consumed in the Low Power Mode (LPM)
3. Receiving power consumption value equals 20.0 multiplies by the energy estimation consumed in the LISTEN Mode.
4. Transmitting power consumption value equals 17.7 multiplies by the energy estimation consumed in the TRANSMIT Mode.

Where (1.8, 0.0545, 20.0, 17.7) represent the level of power consumption of mote in several different hardware states which are expressed in milliamperes (mA). These values are related to the hardware of TMote sky. Thus, CPU requires 1.8 mA, 17.7 mA in transmitting mode, 20.0 mA for receiving mode and requires 0.0545 mA in low power or sleep mode (LPM).

In order to calculate the power consumption in milliwatt per seconds; we should multiply the total consumption by voltage and divide by the number of clock ticks per second which is RTIMER_SECOND. The next section shows the results obtained from implementing these algorithms with CoAP along with the explanation

5. Experimental Results and Evaluation

Devices used in the following experiments runs Contiki operating system [9] which is considered the most suitable operating system for low power devices in the M2M environment, also this operating system is implemented in Cooja simulator. In the next experiments the proposed algorithm and CoAP-RSA algorithm were manipulated tested under the same parameters and conditions. Table 3 shows the simulation parameters used in experiments.

Table 3. Cooja simulation parameters.

Parameters	Values
Operating System	Contiki 3.0
Simulator	Cooja
Nodes Type	Tmote Sky
Physical topologies	1,2,3,4,5 (see section 4.3)
MAC/adaptation layer	ContikiMAC/ 6LowPAN
Routing Protocol	RPL
Radio Environment	Unit Disk Graph Medium (UDGM)
Nodes count	5-320 + RD node
Simulation Duration	Variable
Full Battery	7000 mJ
Transmission Range	50 m

5.1. Results

Figure 3 shows the simulation results for the average level of battery for all nodes to the time of simulation run. As shown, the proposed secure CoAP overcome the CoAP using RSA by 47% in terms of saving energy. Furthermore, when using RSA, the energy of the battery consumed faster than the proposed secure CoAP.

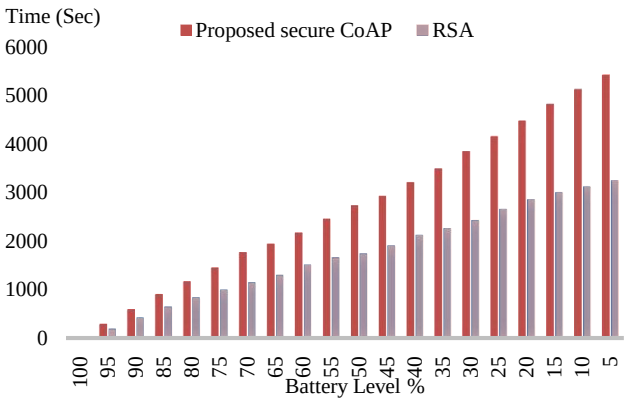

Figure 3. Network lifetime in various battery levels for the two protocols using RSA and ECC algorithms.

The next figures hows the power consumption between proposed secure CoAP and CoAP using RSA when varying the key sizes for both algorithms. As illustrated in the figure, the proposed secure CoAP always consumes less power than in CoAP using RSA which consumes a large amount of power whenever the key size increases. The consumption of power in CoAP using RSA start to be linear when the key size reaches 512 bits.

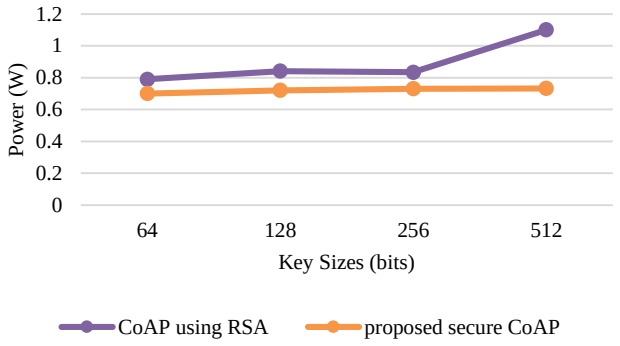

Figure 4. The power consumption between proposed secure CoAP and CoAP using RSA.

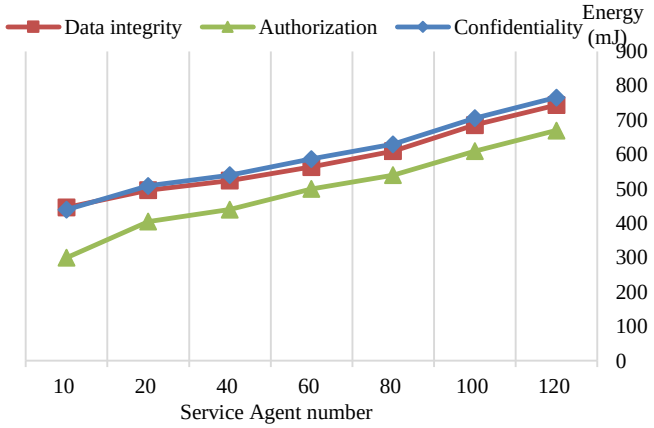

Figure 5. Comparing the three security services when selected in the proposed security-aware CoAP using ECC.

The above figure shows the results that were gathered to show the energy efficiency of the proposed work in this paper. Whereas, the most effective service in terms of energy saving is the authentication when a 75.3% energy savings is noticed. However, both data integrity and confidentiality also saves energy but with 55.7% for data integrity and 47% when confidentiality service is selected.

5.2. Explanation

As mentioned before, the main advantage of ECC is its compact key size because the Elliptic curve often utilizes a smaller key size rather than the other classical equivalent algorithms such as RSA. The variation in identical key sizes can increase whenever the key sizes increase. The next table 4 shows a comparison between the ECC, symmetric algorithms and asymmetric algorithms in terms of security strength similarity.

Table 4. Key size comparison [7].

Symmetric Key Length	Standard asymmetric Key Lenght	ECC Key Lenght
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

As illustrated in the previous table, ECC's key has the smallest sizes that are much less than the other algorithms. Thus, reducing the key size can, in turn, increase the computational speed and at the same time save more energy. Moreover, fast key generation method is a core advantage in ECC. Hence, the processing speed will be increased. As a result, the residual energy of the nodes will be saved and the network lifetime will increase. Using ECC will give the ability to provide authorization, authentication, Integrity and Confidentiality as security services to the IoT networks as follows:

1. *Confidentiality*: using this security service any unauthorized node is denied from accessing the data.
2. *Authorization*: this security service gives each node a unique key pair (public and private) in order to make encryption and decryption. This service is achieved by employing public key, if any malicious/ anonymous node needs to communicate with network nodes then it needs the public key pair of the authorized node.
3. *Integrity*: assurance that messages received by a destination node have not been changed in transit either through collision or via a deliberate tampering by an untrusted node by using a hash chain and MAC list.

6. Conclusions

In this paper a proposed secure and energy efficient CoAP protocol to be in the application layer of the IoT networks that grantees the Confidentiality, Authorization and the Data Integrity security services for the data transferring between the service agents over the IoT network. The proposed protocol uses the

Elliptic Curve Cryptography (ECC) technique as an underlying security Algorithm. The reason for choosing this algorithm is that this algorithm uses smaller key sizes for encryption and decryption which is expected to reduce the power consumption in this type of networks. This work mainly has two parts; the first part implements the CoAP using ECC and using RSA algorithms where the results have proven that using ECC much better than RSA in terms of energy saving. The second part of this paper shows the proposed evaluation function. The second part also focuses on the security services that were applied in the proposed protocol. The results show that authentication achieved a 75.3% energy savings, data integrity had a 55.7% energy saving and confidentiality achieved a 47% energy saving.

References

[1] Ahrary A., Ludena D., Horibe N., and Yang W., "Iot-Security Approach Analysis For The Novel Nutrition-Based Vegetable Production And Distribution System," *in Proceedings of the 3rd International Conference on Advanced Applied Informatics*, Kitakyushu, pp. 185-189, 2014.

[2] Albalas F., Al- Soud M., and Almomani O., "A Proposed Secure And Energy-Effictive Coap Application Layer Protocol For The Internet Of Things," *in Proceedings of the International Arab Conference on Information Technology*, Yassmine Hammamet, 2017.

[3] Alghamdi T., Lasebae A., and Aiash M., "Security Analysis of The Constrained Application Protocol In The Internet Of Things," *in Proceedings of Second International Conference on Future Generation Communication Technology*, London, pp. 163-168, 2013.

[4] Bhattacharyya A., Bose T., Bandyopadhyay S., Ukil A., and Pal A., "LESS: Lightweight Establishment of Secure Session: A Cross-Layer Approach Using CoAP and DTLS-PSK Channel Encryption," *in Proceedings of IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*, Gwangiu, pp. 682-687, 2015.

[5] Brachmann M., Garcia-Morchon O., and Kirsche M., "Security For Practical Coap Applications: Issues And Solution Approaches," Technical Report, 2011

[6] Cao Z., Kovatsch M., Tian H., and He X., "Energy Efficient Implementation of IETF Constrained Protocol Suite Draft-Ietflwig-Energy-Efficient-00," Technical Report, 2014.

[7] Colitti, W., Steenhaut, K., and De Caro, N. "Integrating Wireless Sensor Networks With The Web," *in Proceedings of Extending the Internet to Low power and Lossy Networks*, Chicago, 2011.

[8] Curtis B., "Delivering Security By Design in the Internet of Things," *in Proceedings of the IEEE International Test Conference*, Seattle, pp. 1-1, 2014.

[9] Dunkels, A., http://www.contiki-os.org/, Last Visited, 2017.

[10] Kinney P, http://www.ieee802.org/15/pub/TG4.html, Last Visited, 2016.

[11] Kerasiotis F., Prayati A., Antonopoulos C., Koulamas C., and Papadopoulos G., "Battery Lifetime Prediction Model for a WSN Platform," *in Proceedings of the 4th International Conference on Sensor Technologies and Applications*, Venice, pp. 525-530, 2010.

[12] Kothmayr T., "Security Architecture for Wireless Sensor Networks Based on DTLS," M.S. Thesis, the University of Augsburg, 2011.

[13] Kothmayr T., Schmitt C., Hu W., Brunig M., and Carle G., "DTLS Based Security and Two-Way Authentication for the Internet of Things," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2710-2723, 2013.

[14] Park J. and Kang N., "Lightweight Secure Communication for Coap-Enabled Internet of Things Using Delegated DTLS Handshake," *in Proceedings of the International Conference on Information and Communication Technology Convergence*, Busan, pp. 28-33, 2014.

[15] Rahman A. and Dijk E., "Group Communication for Coap," Technical Report, 2013.

[16] Raza S., Trabalza D., and Voigt T., "6LoWPAN compressed DTLS for CoAP," *in Proceedings of IEEE 8th International Conference on Distributed Computing in Sensor Systems*, Hangzhou, pp. 287-289, 2012.

[17] Raza S., Shafagh H., Hewage K., Hummen R., and Voigt T., Lithe: "Lightweight Secure Coap For The Internet Of Things," *IEEE Sensors Journal*, vol.13, no.10, pp. 3711-3720, 2013.

[18] Ukil A., Bandyopadhyay S., Bhattacharyya A., Pal A., and Bose T. "Lightweight security scheme for IoT applications using CoAP," *International Journal of Pervasive Computing And Comunications*, vol. 10, no. 4, pp. 372-392, 2014.

Firas ALbalas is an Assistant Professor at the Department of Computer Science, Jordan University of Science and Technology, Irbid, Jordan. He received his PhD in Computer Science from Glamorgan (South Wales) University, Cardiff, UK in 2009. His current research interests include Internet of Things, mobile computing, ad hoc networks and wireless sensor networks.

Majd Al-Soud received here M. Sc. in Computer Science with excellence, from Jordan University of Science and Technology, Jordan. She is a research assistant and a part time lecturer at the Department of Computer Science, Jordan University of Science and Technology, Irbid, Jordan. Here current research interests are Computer Networks, Data mining, Information Retrieval and Software Engineering. Before obtaining her Master's degree she has worked as Computer engineer in Bradford and a programmer in Epkss.

Omar Almomani received his PhD degree in computer Science from university Utara Malaysia, Malaysia (2010). He is currently and associate professor at The World Islamic Sciences and Education University (WISE), Amman, Jordan. his current research interests include network performance, network quality of service, wireless sensor networks and Grid computing.

Ammar Almomani received PhD degree from UniversitySains Malaysia (USM) in 2013. He has published more than 45 research papers in International Journals and Conferences of high repute. Currently he is assistant professor and senior lecturer at Dept. of Information Technology, Al-Huson University College, Al-Balqa Applied University, Jordan. His research interest includes advanced Internet security and monitoring.