Hindawi Publishing Corporation
Mobile Information Systems
Volume 2015, Article ID 506284, 11 pages
http://dx.doi.org/10.1155/2015/506284

Research Article

Compact Extensible Authentication Protocol for the Internet of Things: Enabling Scalable and Efficient Security Commissioning

Marcin Piotr Pawlowski,[1,2] Antonio J. Jara,[1,3] and Maciej J. Ogorzalek[2]

[1]*Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), 3960 Sierre, Switzerland*
[2]*Department of Information Technologies, Faculty of Physics, Astronomy and Applied Computer Science, Jagiellonian University, 30-348 Krakow, Poland*
[3]*Research and Development Department, HOP Ubiquitous S.L., Ceuti, 30562 Murcia, Spain*

Correspondence should be addressed to Marcin Piotr Pawlowski; m.p.p@ieee.org

Received 4 August 2015; Revised 1 November 2015; Accepted 4 November 2015

Academic Editor: Laurence T. Yang

Internet of Things security is one of the most challenging parts of the domain. Combining strong cryptography and lifelong security with highly constrained devices under conditions of limited energy consumption and no maintenance time is extremely difficult task. This paper presents an approach that combines authentication and bootstrapping protocol (TEPANOM) with Extensible Authentication Protocol (EAP) framework optimized for the IEEE 802.15.4 networks. The solution achieves significant reduction of network resource usage. Additionally, by application of EAP header compacting approach, further network usage savings have been reached. The EAP-TEPANOM solution has achieved substantial reduction of 42% in the number of transferred packets and 35% reduction of the transferred data. By application of EAP header compaction, it has been possible to achieve up to 80% smaller EAP header. That comprises further reduction of transferred data for 3.84% for the EAP-TEPANOM method and 10% for the EAP-TLS-ECDSA based methods. The results have placed the EAP-TEPANOM method as one of the most lightweight EAP methods from ones that have been tested throughout this research, making it feasible for large scale deployments scenarios of IoT.

1. Introduction

One of disrupting technologies that has a big impact on our lives has been the Internet of Things (IoT) [1, 2]. It has been expected that, by the year 2020, billions of new IoT devices will be connected and deployed around the world [3]. Homes, hospitals, offices, cars, and even cities will be filled with myriads of new devices that will be responsible for wellbeing and safety of its users [4–7]. Consequently, IoT needs to be reliable and easy to use and secure and provide mechanisms for scalable seamless commissioning.

One of the major challenges in the IoT has been the security [8, 9]. High constraints of IoT devices communications and memory and computation capabilities and limited entropy sources [10] in conjunction with the fact that most of devices have been battery operated and have limited remote maintenance capabilities [11] have made security a very challenging goal. Therefore, IoT requires security solutions to be as lightweight as possible and as secure as possible—combination hard to achieve. Furthermore, designed solutions should be easy to use and should not require any human intervention during their lifetime. Additionally, IoT presents new challenges for the bootstrapping and commissioning of billions of deployed devices. Such processes need to be executed without any maintenance time or human intervention.

Many efforts have been carried out by research community addressing security issues of the IoT. The most notable attempts have been coming from the Internet Engineering Task Force (IETF) Datagram Transport Layer Security In Constrained Environments (DICE) [12] working group that has focused on adaptation of the *Transport Layer Security* (TLS) protocol for protection of the end-to-end communication of constrained IoT devices [13]. Additionally, the IETF working group Authentication and Authorization

for Constrained Environments (ACE) has been addressing problems of secure and privacy oriented authorization and authentication in the IoT networks [14].

The main motivation behind this research has been the need to address issues of very lightweight, flexible, scalable, and secure solution for authentication and bootstrapping of constrained devices in the IoT networks. In the environment with overwhelming number of deployed IoT devices new solutions are required to enable secure and scalable management. First step in managing of such enormous number of devices has to be provided already during the deployment phase. The installation and initial set-up of the device need to be seamless and should be done in a place-and-move-along manner. Person responsible for the deployment should only put the device in desired place and walk away: the device should do the rest in a secure way. This is the ideal approach towards which presented solution has been pursuing.

First contribution of this paper has been the combination of *Trust Extension Protocol for Authentication of New deployed Objects and sensors through the Manufacturer* (TEPANOM) bootstrapping and authentication solution with *Extensible Authentication Protocol* (EAP) authentication framework that has been enabled to work over IEEE 802.15.4 networks through *Slim Extensible Authentication Protocol Over Low-Rate Wireless Personal Area Networks* (SEAPOL) adaptation layer [15]. The EAP-TEPANOM solution has achieved substantial reduction of 42% in the number of transferred packets and 35% reduction of the transferred data.

Second contribution of this paper has been the compacted version of the EAP which achieved further minimization of the transmitted data. Application of the EAP header compaction constituted reduction of up to 80% of the EAP header size. That comprises further reduction of up to 10% of transferred data.

The rest of the paper is organized as follows. Section 2 introduces briefly the secure commissioning and current state-of-the-art solutions alongside with the descriptions of the EAP, SEAPOL adaptation layer, and TEPANOM solution. In Section 3 the EAP-TEPANOM solution has been presented. Section 4 presents header compaction solution for the EAP. Section 5 briefly analyzes security of the proposed solutions. The network usage evaluation results have been presented in Section 6. In Section 7 the energy analysis of proposed solutions has been presented. Section 8 concludes the paper and addresses the next research steps.

2. Secure Commissioning

Throughout this paper, the *commissioning* term has been defined as the process of verification and configuration of the deployed equipment. The commissioning consists of two stages, first, the authentication during which deployed devices present their credentials to the network security management mechanism and, second, the bootstrapping stage that is responsible for providing necessary information to deployed devices to enable their functionality, like shared keys negotiation for secure communication.

From the perspective of the security of the commissioning process the description of the state of the art has been limited to the authentication and key establishments mechanisms for the constrained networks.

In [16] the authors present an authentication and key establishment scheme for WSNs in distributed IoT applications. It has been based on a simplified *Datagram Transport Layer Security* (DTLS) [12] exchange and the use of elliptic curve cryptography through TinyECC implementation [17].

One of the most promising bootstrapping security protocols has been the *Host Identity Protocol-Diet EXchange* (HIP-DEX) [18]. Its use for the network access stage has been analyzed in [19]. Although presented results have been promising, compared to the DTLS protocol, the HIP-DEX has not been widely adopted. This has been mainly related to the fact that HIP-DEX does not use certificate-based public key agreement. Additionally, high complexity of the puzzle mechanism responsible for mitigation of the DoS attacks also limits its usability.

Another noteworthy bootstrapping security protocol has been based on the *Protocol for Carrying Authentication for Network Access* (PANA) [20]. In [21] the usage of PANA in networking environment of constrained IoT devices has been presented. The authors showed *PANATIKI* solution which has been a lightweight implementation of PANA that has been suitable for constrained IoT devices. The *Extensible Authentication Protocol-Pre-Shared Key* (EAP-PSK) [22] has been used as the authentication mechanism. Usage of symmetric key cryptography (EAP-PSK) has been motivated by reduction of high computational costs in comparison to the public key cryptography operations. Unfortunately, the EAP-PSK provides lower degree of scalability and security than public key based authentication mechanisms.

All of the above solutions share common issue: they require IP connectivity before the authentication phase. In some cases this might introduce potential security threats. The approach presented in this paper has been based on the link layer and does not require involving higher layer connectivity before authenticating the device.

In the following subsections related works have been briefly described on which the solution presented in this paper has been based.

2.1. EAP over SEAPOL Adaptation Layer. In this subsection basic information about *Extensible Authentication Protocol* (EAP) and its adaptation layer for the IEEE 802.15.4 networks has been provided.

The EAP has been selected as a mechanism for securing the commissioning in IoT networks, based on three observations. First, the EAP has been one of the most commonly used authentication protocols in the Wireless Local Area Networks and it supports many different authentication mechanisms. Second, it works on the link layer; thus it introduces lower communication overhead in comparison to different authentication mechanisms—energy consumption is important for constrained IoT devices. Third, the protocol is flexible and does not require globally centralized infrastructure. Therefore, it provides good scaling capabilities. All of these make the EAP the best choice for secure commissioning solution that combines strong security (like ECC certificates and EAP-TLS authentication method) with lightweight link-layer

transmission and scalable, globally decentralized infrastructure.

The EAP has been a part of the infrastructure specified in the IEEE 802.1X standard [23]. The standard describes comprehensive authentication solution that consists of three services (*Authentication Server*, *Authenticator*, and *Supplicant*), with additional definition of protocol to transfer the EAP frames over the Local Area Networks (EAPOL). Subsequent standard IEEE 802.11i extends the usage of EAPOL protocol over Wireless Local Area Networks [24]. The communication between the Authenticator and the Authentication Server has been realized by the RADIUS protocol and has been defined in [25]; additional protocols such as Diameter also can be used for transporting the EAP Packets [26].

The simple schema showing IEEE 802.1X communication has been presented in Figure 1.

The *Extensible Authentication Protocol* has been defined in the RFC 3748 standard as an authentication framework that provides common functions for the authentication mechanisms [27]. The standard has not defined any authentication mechanism, except MD5 based one, that has been exemplary and totally insecure. The authentication mechanisms have been defined in many subsequent documents like Pre-Shared Key [22] or Transport Layer Security (TLS) [28] based mechanisms. In the EAP nomenclature the authentication mechanisms have been called EAP methods.

The EAP communication consists mostly of the Request-Response datagram exchange between the Authenticator and the Supplicant. During this, at first the authentication method is negotiated. Then, the Request-Response datagram exchange carries the negotiated authentication method data. And, finally, the authentication procedure one final Success or Failure datagram is sent to the Supplicant.

2.2. SEAPOL-IEEE 802.15.4 Adaptation Layer. The EAP has its link layer transmission mechanisms defined for IEEE 802.1 wired [23] and IEEE 802.11 wireless [24] networks. Officially, such mechanism has not been defined for the IEEE 802.15.4 standard [29], although recently in [30] the EAPOL protocol has been adopted for the needs of the IEEE 802.15.4 link layer [29] and subsequently optimized as *Slim Extensible Authentication Protocol Over Low-Rate Wireless Personal Area Networks* (SEAPOL).

The improved version of the EAPOL protocol has been designed after careful analysis of the regular EAPOL protocol. It has been noticed that only 5 different frame types can represent the full functionality of the regular EAPOL protocol. Additionally the EAPOL Start and EAP Packet frames can be easily differentiated by the frame payload size, and therefore they can use the same frame type. That led to the definition of the *Slim EAPOL* (SEAPOL) that represents full EAPOL functionality in just 3 bits (93.75% less overhead in comparison to the regular EAPOL) and, additionally, it has been fully integrated with the Frame Control field of the IEEE 802.15.4 protocol reducing the frame overhead to zero. The IEEE 802.15.4 Frame Control field modifications of SEAPOL protocol have been presented in Figure 2.

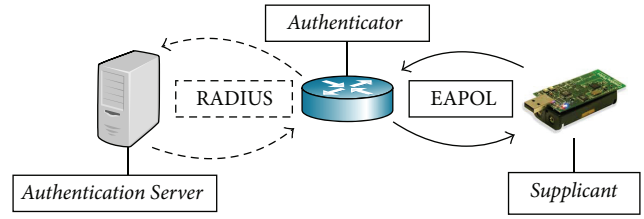


Figure 1: Schema of IEEE 802.1X secured networks architecture.

2.3. TEPANOM Protocol. Trust Extension Protocol for Authentication of New deployed Objects and sensors through the Manufacturer (TEPANOM) has been defined as a solution for authentication, identity verification, bootstrapping, configuration, and trust extension of the deployment and management domains to the new device [31]. The TEPANOM protocol consists of two phases, the *Authentication* and the *Trust Extension*.

2.3.1. Trust Extension. The *Trust Extension* phase of the TEPANOM protocol has been designed to register methods and resources of the new device and to establish new shared key between the protocol actors. In this paper the *Trust Extension* phase of the TEPANOM protocol will be not addressed any further, and therefore for more details please refer to [31].

2.3.2. Authentication. The *Authentication* phase of the TEPANOM protocol has been designed to authenticate the device and its features to the manufacturer through the *TEPANOM-Authentication-Point*. From the perspective of the network communication, three different actors have been defined, the *TEPANOM-Client*, *TEPANOM-Gateway*, and already mentioned *TEPANOM-Authentication-Point*. The *TEPANOM-Client* is a constrained IoT device that is authenticating to the *TEPANOM-Authentication-Point*. The authentication process is done through the *TEPANOM-Guard* that is the gateway between the unauthenticated devices and the privileged parts of the network. The *TEPANOM-Guard* has been responsible for protecting the *TEPANOM-Authentication-Point* against Denial-of-Service attacks, which could have been executed by malicious *TEPANOM-Clients*. The *TEPANOM-Authentication-Point* has been responsible for authenticating the *TEPANOM-Client* and providing the *DataSheet* which is extended description of device resources, capabilities, and methods.

3. EAP-TEPANOM

It has been widely known fact that the most energy consumption in the constrained device comes from the radio communication. Therefore, it has been imperative to minimize the usage of wireless interfaces by limiting the number of transmitted bytes and packets. This approach has been seen in the application layer as CoAP protocol for HTTP [32], for device management like COMAN for SNMP [33, 34] or OMA LWM2 M [35], and for network layer transmission protocol like 6LoWPAN for IPv6 [36], MIPv6 [37, 38], and many others.

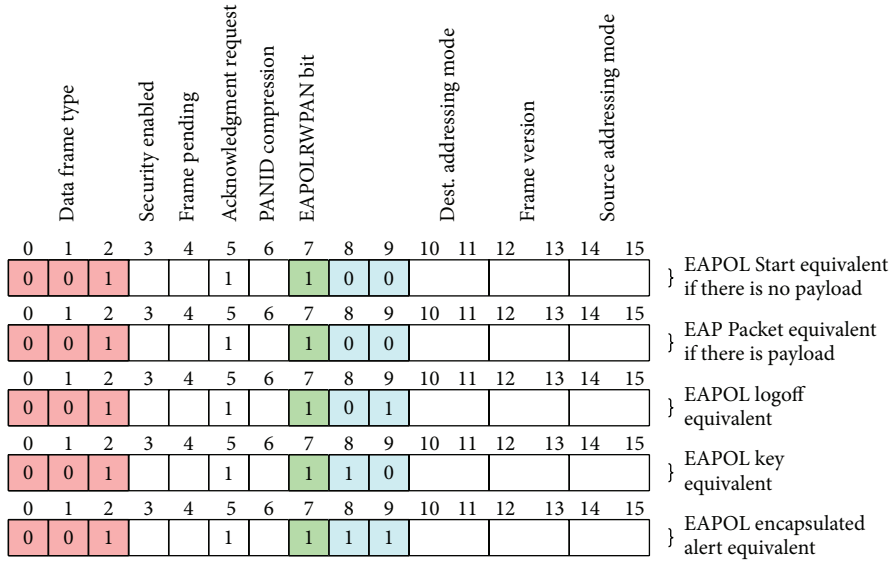


Figure 2: IEEE 802.15.4 Frame Control field modifications to support Slim Extensive Authentication Protocol Over Low-Rate Wireless Personal Area Networks (SEAPOL).

The *EAP-TEPANOM* protocol is a combination of *TEPANOM* with *EAP*. The EAP has been used as a transport layer for the TEPANOM authentication mechanism. This formed new method that has been examined and compared in terms of the IEEE 802.15.4 network resource usage. The main motivation for this approach has been the need to minimize the communication overhead by eliminating the usage of IPv6/UDP transport protocols and exchanging it with more lightweight EAP communication stack.

3.1. Protocol Selection. The EAP-TEPANOM approach maximizes the size of the available payload space in the IEEE 802.15.4 frame by removing the UDP and IPv6 encapsulations that have been used by the regular TEPANOM protocol and using EAP encapsulation instead. The UDP and IPv6 together require 48 bytes of the 127-byte IEEE 802.15.4 frame which constitutes 37.8% of the whole frame. Using the EAP encapsulation with SEAPOL adaptation layer the same task can be achieved by only 5 bytes, which has been only 3.9% of the IEEE 802.15.4 frame. By applying this approach, it has been possible to save additional 43 bytes for the payload. This constitutes the reduction of the number of transmitted and received bytes and packets. Therefore, it has been the main contributing factor to the minimization of the network usage. The visual representation of TEPANOM frame with UDP/IPv6 has been presented in Figure 3(a) and EAP encapsulation of the TEPANOM protocol has been presented in Figure 3(b).

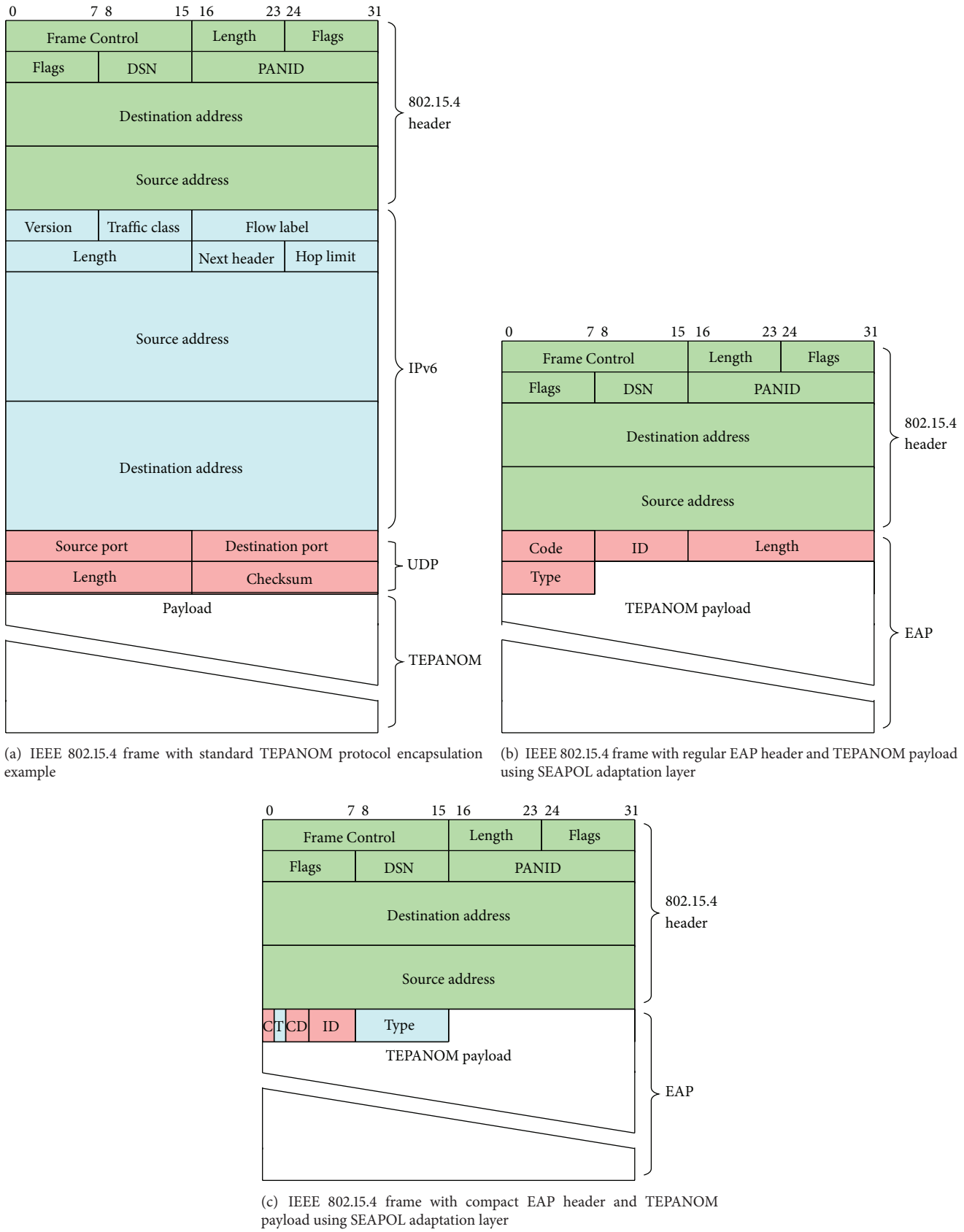
3.2. Communication Exchange. EAP based solution that has been implemented in previous researches [39, 40] has been relying on the usage of the RADIUS protocol as the communication mechanism between the *Authenticator* and the *Authentication Server*. The solution has been extended to enable support for the TEPANOM protocol. This has been achieved by making modifications to the *Authenticator* EAP Packet processing mechanism. Additional functionalities have been introduced that have been responsible for recognizing the EAP-TEPANOM protocol datagrams, extracting the TEPANOM payload, sending the TEPANOM payload to the *TEPANOM-Guard* through UDP/IPv6, and receiving the answers from the *TEPANOM-Guard* and forwarding them to the *Supplicant (TEPANOM-Client)* encapsulated in the EAP datagram. In other words, introduced modifications enabled the *Authenticator* to work as a relay between EAP (*Supplicant*) and UDP/IPv6 (*TEPANOM-Guard*) protocols. The whole communication scheme with mentioned changes has been presented in Figure 4.

4. Compact EAP

The minimization of transferred data for constrained IoT devices has been very important research objective. If a constrained device sends or receives data through the wireless interface it consumes a significant amount of energy. Due to the fact that constrained devices are mostly battery operated it has been imperative to save this limited resource. Throughout this and previous researches and various experiments with the EAP it has been noticed that it is possible to introduce modifications to the EAP header to achieve reduction of its length by up to 80%. Such modification should have significant impact on reduction of the energy consumption of constrained IoT devices.

The compact EAP (cEAP) solution has been presented and discussed in this section. Every field of the EAP header has been discussed separately to show the motivation and reasoning behind the compaction approach.

4.1. Code. First field of the EAP header has been the *Code* whose length is one byte. Possible values that can be put in



(a) IEEE 802.15.4 frame with standard TEPANOM protocol encapsulation example

(b) IEEE 802.15.4 frame with regular EAP header and TEPANOM payload using SEAPOL adaptation layer

(c) IEEE 802.15.4 frame with compact EAP header and TEPANOM payload using SEAPOL adaptation layer

Figure 3: Comparison of IEEE 802.15.4 frames with different encapsulations of TEPANOM protocol.

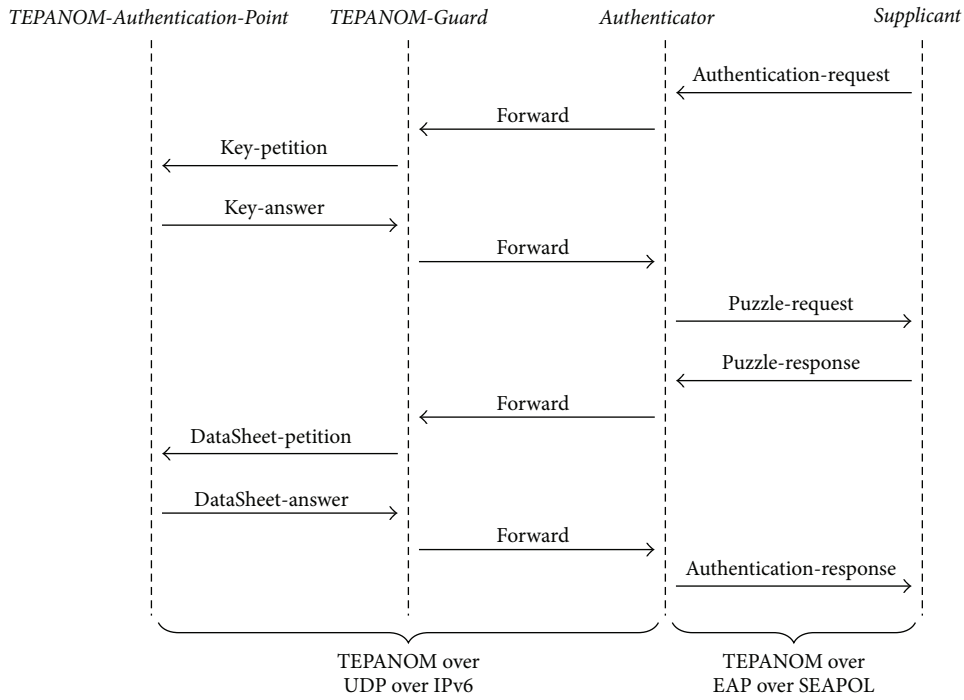


Figure 4: EAP-TEPANOM method message exchange scheme.

the *Code* field have been limited to just four different states: *Request (1)*, *Response (2)*, and *Success (3)* or *Failure (4)*. This shows the obvious fact that the *Code* field can be used more efficiently. By application of simple compaction,

$$\text{Compact}(\text{Code}) = \text{Code} - 1, \tag{1}$$

the *Code* can freely fit into just 2 bits of the cEAP header instead of occupying whole byte of the EAP header. The process is easily reversible by applying

$$\text{Decompact}(\text{Code}) = \text{Code} + 1 \tag{2}$$

after which the *Code* will go back to the original EAP *Code* value.

Application of this code compaction approach makes it possible to save 6 bits out of one byte of the header without compromising any functionality.

4.2. Identifier. Second field of the EAP header as well as second byte has been devoted to the *Identifier*. This field has been introduced as a method to differentiate between Request and retransmission during the EAP session. The new Request needs different *Identifier* than the previous Request. The *Identifier* has been also responsible for the minimization of the possibility of successful reply attack. In addition in the RFC 3748 [27] it has been stated that *Identifier* space is unique to each session. Authenticators are not restricted to only 256 simultaneous authentication conversations, and conversation is not limited to only 256 roundtrips.

Therefore, there has been no significant requirement to have the *Identifier* field length of one byte by applying simple compaction:

$$\text{Compact}(\text{Identifier}) = \text{Identifier} \pmod{16}. \tag{3}$$

It has been possible to shorten the *Identifier* in half and use just 4 bits out of one byte.

The *Identifier* compaction function loses information of the original value. Thus, the decompaction in cases where *Identifier* is not fitting in four least significant bits of the original field would require having saved *Identifier*. This may not be common situation and might only occur if there would be communication between EAP and cEAP.

4.3. Length. Third field of the EAP header and third and fourth bytes have been devoted to the *Length* indicator. This has been the longest field in the whole EAP header. One important thing, which has been noticed during the research on EAP for IoT, has been that the *Length* field has not been used in any significant manner. Additionally, due to the fact that in the IEEE 802.15.4 networks the length of the frame has been limited to 127 bytes and that throughout the research the EAP frame has been limited to 100 bytes there has been no need for the *Length* field. The length of the IEEE 802.15.4 frame has always been sent to the MAC layer which independently indicates the EAP frame size.

Therefore it has been possible to completely remove the *Length* field, saving two bytes of the header.

4.4. Type. Fourth and last field of the EAP header of the length of one byte has been the *Type*. The presence of the *Type* field has been conditional. It is part of the header only

if the *Code* field has been set as the *Request* or *Response*. Observations of the EAP communication have led to the conclusion that the *Type* field could be used in a more conditional manner. In addition to setting proper *Code*, the *Type* field could be only present if the type of the frame needs to be changed during current EAP session. This means that, for example, during the EAP-TEPANOM conversation the *Type* will be set and present in the header only once for the first EAP-TEPANOM packet.

That approach has saved one additional byte per every packet sent during the conversation of the same EAP method, except the first packet of the conversation.

4.5. Complete Solution. The modifications of the EAP fields presented above have been integrated and formed complete solution. After applying the *Code* compaction, the first byte of the header there will have 6 bits free that can be utilized more efficiently.

Firstly, the most-left bit of the first byte of the frame indicates if the EAP header has been compacted. If the most-left bit of the first byte of the frame is turned on then the header has been compacted. This approach will not break any EAP implementation because the RFC 3748 [27] clearly states that any packet with *Code* above the value of 4 should be discarded.

Secondly, the second most-left bit of the first byte of the frame indicates if the *Type* field has been present. If the second most-left bit has been turned on then the *Type* field has been present. If it has been turned off then the *Type* field has been absent.

Next, third and fourth most-left bits have been reserved for the compacted *Code* values. The last field of the cEAP header has been devoted to the compacted *Identifier*.

The presented complete solution has reduced the 5 bytes of regular EAP header down to just 1 byte, achieving 80% of header space savings. The examples of the compacted EAP headers have been presented in Figure 5.

5. Security Analysis

In this work two contributions have been introduced and in this section their brief security analysis has been presented.

The EAP-TEPANOM method has been based on the TEPANOM protocol which has been designed as the lightweight protocol for bootstrapping of constrained devices. Firstly, the TEPANOM protocol itself has been well designed and it has based its security on the AES encryption, which stands its security on the same level as the EAP-PSK method. Secondly, it implements Denial-of-Service mitigation technique through puzzle solving request, which is unique feature among other EAP methods. Lastly, the TEPANOM message transmission has been based on the UDP/IPv6 protocols that have been transparent for internal workings of the TEPANOM protocol and by exchanging UDP/IPv6 with EAP the communication scheme has not been altered. Therefore, EAP-TEPANOM protocol provides security on the level of EAP-PSK, with Denial-of-Service mitigation technique and without UDP/IPv6 overhead.

The security of the compacted EAP has been on the same level as the regular EAP. Although the length of the identifier field has been reduced and its purpose has been to reduce the replay attacks, it should not pose any significant threat due to the fact that in constrained networks number of transmissions is limited. Caution should be taken while using compact EAP in regular networks where number of transmissions is virtually unlimited; probability of a replay attack might be higher. No other changes have been introduced to the functionality of the compacted EAP that could hamper its security.

6. Results

In this section experimental results of the analysis of the EAP methods and compact EAP have been presented. The tests have been performed using TelosB [41] compatible devices working under control of the ContikiOS [42] operating systems. The results have been obtained from the perspective of network resource consumption of the Supplicant (TEPANOM-Client) device. Whole comparison has been presented in Table 1 and visual comparison between regular and compact EAP has been presented in Figure 6.

6.1. Transmission. Both of the TEPANOM and EAP-TEPANOM solutions require only 2 packets to be sent by the Supplicant. This has been the most minimal requirement from all of the previously evaluated EAP methods; even the most simple of the regular EAP methods the EAP-MD5 requires one additional packet to be transmitted by the authenticating device.

There has been no difference in terms of transmitted number of packets between the EAP and compact EAP solutions.

Data required to be sent by the Supplicant have been different for the TEPANOM and EAP-TEPANOM solutions. The EAP-TEPANOM transmits only 98 bytes which has been 53% less than the 210 bytes required by the TEPANOM. The EAP-TEPANOM result has been 33% bigger than the EAP-MD5 method and 46% smaller than the EAP-PSK method. The TEPANOM result has been placed between the results of EAP-PSK and EAP-TLS-ECDSA-160 methods.

The differences between EAP and compact EAP in terms of transmitted bytes by the Supplicant device have been very significant. The number of transmitted bytes for the cEAP-TEPANOM method has been lower by 8.16%. This result has been the smallest from all of the analyzed methods. The biggest reduction has been obtained for the cEAP-TLS-RSA-2048 method and has been 24.15% smaller than the EAP counterpart.

6.2. Reception. The number of received packets has been significantly higher than the number of transmitted packets for both of the TEPANOM and EAP-TEPANOM solutions. The TEPANOM requires receiving 17 packets, which has been the same number of received packets as for the EAP-TLS-ECDSA-160 method. The EAP-TEPANOM requires only 9 packets to be received, which has been 47% less than the TEPANOM. The EAP-TEPANOM results have been placed

Table 1: Comparison of network usage statistics calculated on the Supplicant/Client node. The statistics have been generated for compact (cEAP) and standard header versions of various EAP methods and regular TEPANOM protocol encapsulation. All statistics have been generated using SEAPOL adaptation layer for the IEEE 802.15.4 network.

	TX packets	TX data	RX packets	RX data	Total packets	Total data
TEPANOM	2	210 B	17	1533 B	19	1743 B
EAP-TEPANOM	2	98 B	9	1020 B	11	1118 B
cEAP-TEPANOM	2	90 B	9	985 B	11	1075 B
EAP-MD5	3	66 B	3	59 B	6	125 B
cEAP-MD5	3	54 B	3	49 B	6	103 B
EAP-PSK	5	181 B	4	160 B	9	341 B
cEAP-PSK	5	161 B	4	145 B	9	306 B
EAP-TLS-ECDSA-160	12	271 B	17	812 B	29	1083 B
cEAP-TLS-ECDSA-160	12	223 B	17	745 B	29	960 B
EAP-TLS-ECDSA-256	13	286 B	18	931 B	31	1217 B
cEAP-TLS-ECDSA-256	13	234 B	18	860 B	31	1094 B
EAP-TLS-RSA-480	19	376 B	24	1566 B	43	1942 B
cEAP-TLS-RSA-480	19	300 B	24	1471 B	43	1771 B
EAP-TLS-RSA-512	20	397 B	25	1627 B	45	2024 B
cEAP-TLS-RSA-512	20	317 B	25	1528 B	45	1845 B
EAP-TLS-RSA-1024	27	496 B	32	2370 B	59	2866 B
cEAP-TLS-RSA-1024	27	388 B	32	2243 B	59	2631 B
EAP-TLS-RSA-2048	43	712 B	48	4200 B	91	4912 B
cEAP-TLS-RSA-2048	43	540 B	48	4009 B	91	4549 B

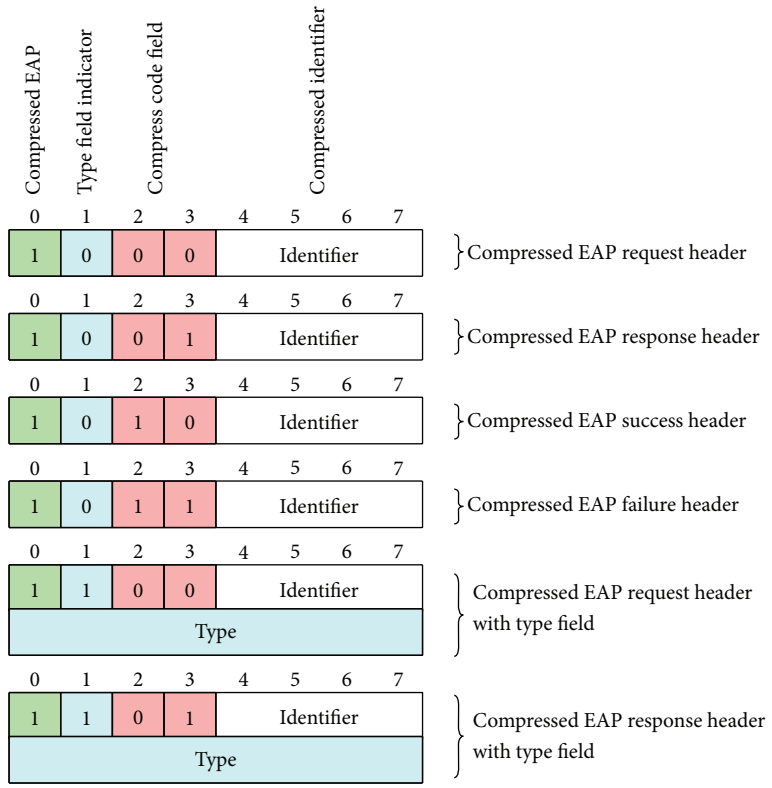


Figure 5: Compact EAP header examples.

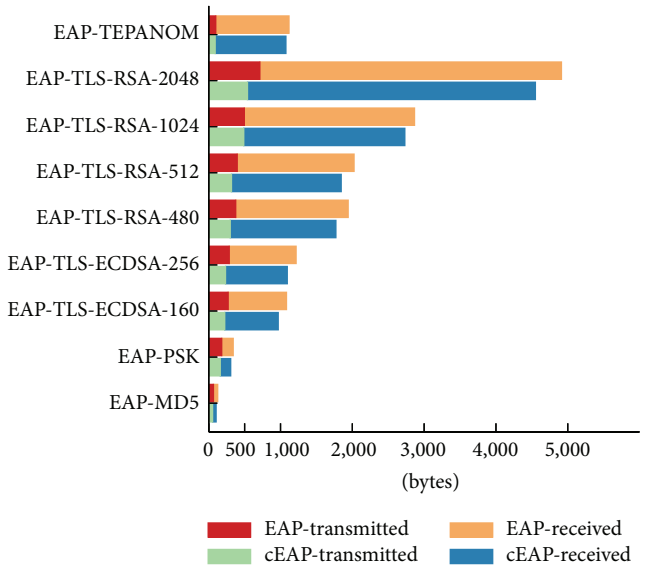


Figure 6: Comparison of network usage calculated on the Supplicant/Client node of regular and compacted EAP using various EAP methods with SEAPOL adaptation for the IEEE 802.15.4 network.

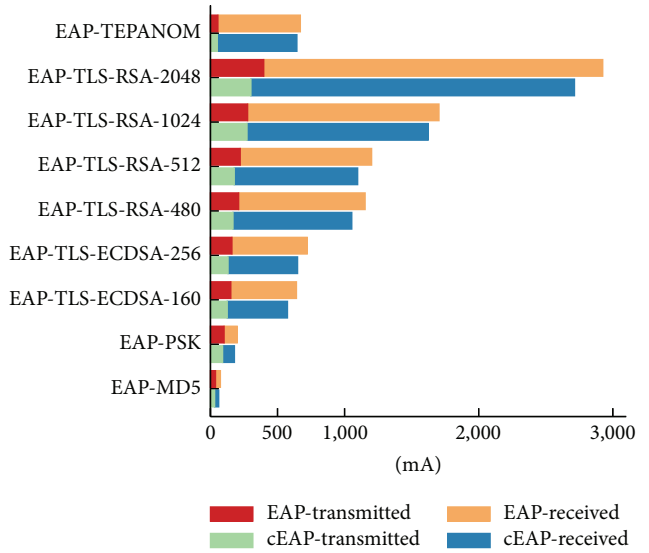


Figure 7: Comparison of estimated energy consumption calculated on the Supplicant/Client node of regular and compacted EAP using various EAP methods with SEAPOL adaptation for the IEEE 802.15.4 network.

between the results of the EAP-PSK and EAP-TLS-ECDSA-160 methods.

There has been no difference in terms of received number of packets between the EAP and compact EAP solutions.

Received data has risen up significantly in comparison to the transmitted data for both of the TEPANOM and EAP-TEPANOM solutions. The TEPANOM protocol needs to receive 1533 bytes of data, which makes it almost the same result as for the EAP-TLS-RSA-480 method. The EAP-TEPANOM needs to receive 33% less data than the TEPANOM, which has been 1020 bytes. These results make the EAP-TEPANOM just slightly more data hungry than the EAP-TLS-ECDSA-256 solution.

The differences between EAP and compact EAP in terms of received bytes by the Supplicant device have been less significant than those of the transmitted bytes. For the cEAP-TEPANOM the reduction has been at the level of 3.43%, which also has been the smallest improvement from all of the measured methods. The biggest reduction has been measured for the cEAP-MD5 method and has been at the level of 16.94% in comparison to the regular EAP.

6.3. Total. The total number of packets for the TEPANOM protocol has been 19 and for the EAP-TEPANOM 11 that has a 42% reduction. This places both solutions between the EAP-PSK and EAP-TLS-ECDSA-160 methods results.

There has been no difference in terms of total number of packets between the EAP and compact EAP solutions.

The total number of received data for the TEPANOM protocol has been 1743 bytes and for EAP-TEPANOM has been 35% less, which has been 1118 bytes. These results have placed the TEPANOM between EAP-TLS-ECDSA-256 and EAP-TLS-RSA-480 methods and the EAP-TEPANOM has been placed betweeen EAP-TLS-ECDSA-160 and EAP-TLS-ECDSA-256 results.

The total reduction of exchanged data during the compact EAP communication has been significant. For the cEAP-TEPANOM method savings have been at the level of 3.84% and this has been the smallest saving from all of the analyzed methods. The biggest savings have been measured for the cEAP-MD5 method and have been at the level of 17.6%. Noteworthy savings have been achieved for the cEAP-TLS-ECDSA-160 and for the cEAP-TLS-ECDSA-256 at the level of 10.61% and 10.10%, respectively.

7. Energy Analysis

In this section the energy analyses of introduced solutions have been presented. During the analysis the TelosB compatible mote has been used equipped with *TI MSP430F1611 Microcontroller* and *CC2420 Radio Chip*. For the purpose of the analysis the energy consumption models from [43] have been used.

The CC2420 radio chip requires on average 17.4 mA for transmission and 18.8 mA for reception of packet [44]. Using these numbers the energy analysis of our solution has been prepared and presented in Figure 7.

Application of compacted EAP solution saves up to 24.1% while transmitting EAP-TLS-RSA-2048 authentication data and 4.55% while receiving for the same method. This gives total saving of 7.2%. The biggest total saving of energy of 17.11% has been observed for EAP-MD5 method. The EAP-PSK and EAP-TLS-ECDSA have achieved energy savings of around 10%, while smallest reduction of the energy consumption of 3.81% has been noted for EAP-TEPANOM method.

Overall, application of compacted EAP brings on average 8.96% savings of energy consumption.

8. Conclusion and Future Work

First contribution in this work has been a solution that combines the *Extensive Authentication Protocol* (EAP) with *Slim Extensive Authentication Protocol Over Low-Rate Wireless Personal Area Networks* (SEAPOL) IEEE 802.15.4 adaptation layer with *Trust Extension Protocol for Authentication of New deployed Objects and sensors through the Manufacturer* (TEPANOM). This solution has been evaluated and achieved significant network usage savings. The EAP-TEPANOM method has achieved 42% reduction in number of transferred packets and 35% reduction of the data that needs to be transferred. The EAP-TEPANOM has been requiring fewer network resources than the most of the EAP-TLS methods.

Second contribution is that the compaction of the EAP header has been proposed that achieved significant savings in transmitted and received data. The best overall savings have been obtained for the cEAP-MD5 method at the level of 17.6% and for the cEAP-TLS-ECDSA based ones at the level of 10%. The cEAP-TEPANOM method has achieved saving at the level of 3.84%.

The EAP-TEPANOM solution showed that it has been possible to use the EAP infrastructure to reduce the usage of the network resources of the constrained devices and extend it to communicate with new authentication protocols and its separate infrastructure.

Future work will be devoted to integrating more closely the TEPANOM solution and its architecture with the EAP infrastructure. More work will be done in the context of the TEPANOM Trust Extension phase integration with EAP infrastructure and its optimization. Additionally the EAP will be analyzed more thoroughly and new approach would be designed to find better way to transmit data more efficiently by reducing the number of transmitted packets.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This project has been supported by the Swiss national government through the Sciex-NMSch (Scientific Exchange Programme between Switzerland and the New Member States of the EU) with the Project Code 13.121, named BASTION "Bootstrapping, Authentication, Security and Trust for the Internet of Things Networks." The authors would like to thank also projects SAFESENS ENIAC Joint Undertaking with the Grant Agreement no. 621272 and the EU Horizon 2020 projects ENTROPY with the Grant Agreement no. 649849 and INPUT with the Grant Agreement no. 644672.

References

[1] O. Vermesan, P. Friess, P. Guillemin et al., "Internet of things strategic research roadmap," in *Internet of Things: Global Technological and Societal Trends*, O. Vermesan, P. Friess, P. Guillemin et al., Eds., vol. 1, pp. 9–52, 2011.

[2] International Telecommunication Union, "The internet of things—executive summary," iTU Internet Reports, 2005.

[3] K. Pretz, *The Next Evolution of the Internet*, IEEE Magazine, 2013.

[4] A. S. Taylor, R. Harper, L. Swan, S. Izadi, A. Sellen, and M. Perry, "Homes that make us smart," *Personal and Ubiquitous Computing*, vol. 11, no. 5, pp. 383–393, 2007.

[5] D. Niyato, E. Hossain, and S. Camorlinga, "Remote patient monitoring service using heterogeneous wireless access networks: architecture and optimization," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 4, pp. 412–423, 2009.

[6] A. Clarke and R. Steele, "Health participatory sensing networks," *Mobile Information Systems*, vol. 10, no. 3, pp. 229–242, 2014.

[7] H. Chen and Z. Fu, "Hadoop-based healthcare information system design and wireless security communication implementation," *Mobile Information Systems*, vol. 2015, Article ID 852173, 9 pages, 2015.

[8] T. Heer, O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar, and K. Wehrle, "Security challenges in the ip-based internet of things," *Wireless Personal Communications*, vol. 61, no. 3, pp. 527–542, 2011.

[9] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed internet of things," *Computer Networks*, vol. 57, no. 10, pp. 2266–2279, 2013.

[10] M. Pawlowski, A. Jara, and M. Ogorzalek, "Harvesting entropy for random number generation for internet of things constrained devices using on-board sensors," *Sensors*, vol. 15, no. 10, pp. 26838–26865, 2015.

[11] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: challenges, design principles, and technical approaches," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4258–4265, 2009.

[12] E. Rescola and N. Modadugu, "Rfc 4347: Datagram transport layer security (dtls)," Request for Comments, IETF, 2006.

[13] DICE Working Group, "IETF DTLS in Constrained Environments (DICE) Working Group," https://datatracker.ietf.org/wg/dice/charter/.

[14] ACE Working Group, "IETF Authentication and Authorization for Constrained Environments (ACE) Working Group," https://datatracker.ietf.org/wg/ace/charter/.

[15] M. P. Pawlowski, A. J. Jara, and M. J. Ogorzalek, "Eap for iot: more efficient transport of authentication data—tepanom case study," in *Proceedings of the 29th IEEE International Conference on Advanced Information Networking and Applications Workshops (AINA '15)*, pp. 694–699, IEEE, Gwangiu, The Republic of Korea, March 2015.

[16] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "PAuthKey: a pervasive authentication protocol and key establishment scheme for wireless sensor networks in distributed iot applications," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 357430, 14 pages, 2014.

[17] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 245–256, IEEE, St. Louis, Mo, USA, April 2008.

[18] R. Hummen, J. Hiller, M. Henze, and K. Wehrle, "Slimfita hip dex compression layer for the ip-based internet of things," in *Proceedings of the IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '13)*, pp. 259–266, 2013.

[19] O. Garcia-Morchon, S.-L. Keoh, S. S. Kumar, P. Moreno-Sanchez, F. Vidal-Meca, and J. H. Ziegeldorf, "Securing the IP-based internet of things with HIP and DTLS," in *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '13)*, pp. 119–124, ACM, Budapest, Hungary, April 2013.

[20] D. Forsberg, Y. Ohba, B. Patil, H. Tschofenig, and A. Yegin, "Rfc 5191 protocol for carrying authentication for network access (pana)," Network Working Group, 2008.

[21] P. M. Sanchez, R. M. Lopez, and A. F. G. Skarmeta, "PANATIKI: a network access control implementation based on PANA for IoT devices," *Sensors*, vol. 13, no. 11, pp. 14888–14917, 2013.

[22] F. Bersani and H. Tschofenig, "Rfc 4764-the EAP-PSK protocol: a pre-shared key extensible authentication protocol (EAP) method," Tech. Rep., IETF Network Working Group, 2007.

[23] IEEE LAN/MAN Standards Committee, "IEEE Std 802.1 X-2004 (Revision of IEEE Std 802-1x-2001)," 2004.

[24] LAN/MAN Standards Committee, "IEEE 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements," 2004.

[25] B. Aboba and P. R. Calhoun, "Radius (remote authentication dial in user service) support for extensible authentication protocol (EAP)," 2003.

[26] G. Zorn, T. Hiller, and P. Eronen, "Diameter extensible authentication protocol (eap) application," 2005.

[27] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible authentication protocol (eap)," Tech. Rep. RFC 3748, 2004.

[28] D. Simon, B. Aboba, R. Hurst, and D. Simon, "Rfc 5216 the eap-tls authentication protocol," 2008.

[29] IEEE LAN/MAN Standards Committee, *IEEE Standard for Local and Metropolitan Area Networks—Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Computer Society, 2011.

[30] M. P. Pawlowski, A. J. Jara, and M. J. Ogorzalek, ""Extending extensible authentication protocol over," in *Proceedings of the 8th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IEEE 802.15.4 Networks (IMIS '14)*, pp. 340–345, Birmingham, UK, July 2014.

[31] A. J. Jara, "Trust extension protocol for authentication in networks oriented to management (TEPANOM)," in *Availability, Reliability, and Security in Information Systems: IFIP WG 8.4, 8.9, TC 5 International Cross-Domain Conference, CD-ARES 2014 and 4th International Workshop on Security and Cognitive Informatics for Homeland Defense, SeCIHD 2014, Fribourg, Switzerland, September 8–12, 2014. Proceedings*, vol. 8708 of *Lecture Notes in Computer Science*, pp. 155–165, Springer, 2014.

[32] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, *Constrained Application Protocol (COAP), Draft-IETF-Core-Coap-13*, The Internet Engineering Task Force (IETF), Orlando, Fla, USA, 2012.

[33] M. Ersue, D. Romascanu, J. Schoenwaelder, and U. Herberg, *Management of Networks with Constrained Devices: Problem Statement and Requirements*, 2015.

[34] M. Ersue, D. Romascanu, J. Schoenwaelder, and A. Sehgal, *Management of Networks with Constrained Devices: Use Cases*, 2015.

[35] L. Tian, *Lightweight M2M (OMA LWM2M)*, OMA Device Management Working Group (OMA DM WG), Open Mobile Alliance (OMA), 2012.

[36] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of ipv6 packets over ieee 802.15. 4 networks," Internet Proposed Standard RFC 4944, 2007.

[37] A. J. Jara, D. Fernandez, P. Lopez, M. A. Zamora, and A. F. Skarmeta, "Lightweight MIPv6 with IPSec support," *Mobile Information Systems*, vol. 10, no. 1, pp. 37–77, 2014.

[38] I. You, J.-H. Lee, Y. Hori, and K. Sakurai, "Enhancing MISP with fast mobile IPv6 security," *Mobile Information Systems*, vol. 7, no. 3, pp. 271–283, 2011.

[39] J. L. H. Ramos, M. P. Pawlowski, A. J. Jara, A. F. Skarmeta, and L. Ladid, "Toward a lightweight authentication and authorization framework for smart objects," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 4, pp. 690–702, 2015.

[40] L. Marin, M. P. Pawlowski, and A. Jara, "Optimized ECC implementation for secure communication between heterogeneous IoT devices," *Sensors*, vol. 15, no. 9, pp. 21478–21499, 2015.

[41] "TelosB Brekley site," http://openwsn.berkley.edu/wiki/TelosB.

[42] ContikiOS project, http://www.contiki-os.org/.

[43] E. Casilari, J. M. Cano-García, and G. Campos-Garrido, "Modeling of current consumption in 802.15.4/zigbee sensor motes," *Sensors*, vol. 10, no. 6, pp. 5443–5468, 2010.

[44] Texas Instruments, "2.4 GHz IEEE 802.15.4/ZigBee-ready RF Transceiver," http://www.ti.com/lit/ds/symlink/cc2420.pdf.