Lightweighted and energy-aware MIKEY-Ticket for e-health applications in the context of internet of things

Mohammed Riyadh Abdmeziem, Djamel Tandjaoui, Imed Romdhani

Lightweighted and Energy-Aware MIKEY-Ticket For E-Health Applications in the Context of Internet of Things

Mohammed Riyadh Abdmeziem

LSI laboratory, Department of Computer Science,
University of Sciences and Technology Houari Boumedienne (USTHB),
Bab Ezzouar, Algiers, Algeria
E-mail: rabdmeziem@usthb.dz

Djamel Tandjaoui

Security division, Center for Research on Scientific and Technical Information (CERIST), Ben Aknoun, Algiers, Algeria
E-mail: dtandjaoui@mail.cerist.dz

Imed Romdhani

School of Computing, Edinburgh Napier University,
Edinburgh, United Kingdom (UK) E-mail: I.Romdhani@napier.ac.uk

Abstract: E-health applications have emerged as a promising approach to provide unobtrusive and customizable support to elderly and frail people based on their situation and circumstances. However, due to limited resources available in such systems and data privacy concerns, security issues constitute a major obstacle to their safe deployment. To secure e-health communications, key management protocols play a vital role in the security process. Nevertheless, current e-health systems are unable to run existing standardized key management protocols due to their limited energy power and computational capabilities. In this paper, we introduce two solutions to tailor MIKEY-Ticket protocol to constrained environments. Firstly, we propose a new header compression scheme to reduce the size of MIKEY's header from 12 Bytes to 3 Bytes in the best compression case. Secondly, we present a new exchange mode to reduce the number of exchanged messages from six to four. We have used a formal validation method to evaluate and validate the security properties of our new tailored MIKEY-Ticket protocol. In addition, we have evaluated both communication and computational costs to demonstrate the energy gain. The results show a decrease in MIKEY-Ticket overhead and a considerable energy gain without compromising its security properties.

Keywords: E-health; Internet of Things; MIKEY-Ticket; Security; Key Management; Data Confidentiality

Biographical notes: Mohammed Riyadh ABDMEZIEM is a Postdoctoral Researcher at Loria-Inria (University of Lorraine, France). He received his PhD in computer science from the University of Sciences and Technology Houari Boumedienne (Algeria) in 2016. He has been awarded his Master degree in Software Engineering as the top of the 2012 class from the same university. His research activities are focused on security, confidentiality, and key management protocols. Dr. ABDMEZIEM managed to publish several articles at an international level in conference proceedings, book chapters, and journals.

Djamel TANDJAOUI is a Researcher at the Center for Research on Scientific and Technical Information (CERIST) in Algiers, Algeria since 1999. He received his PhD degree from the university of Science and Technology Houari Boumediene (USTHB), Algiers in 2005. He obtained a master degree and an engineer degree in computer science from the same university. At present, he is member of computer security research division at CERIST. His research interest includes mobile networks, mesh networks, sensor networks, ad hoc networks, QoS and security.

Imed ROMDHANI is an Associate Professor at Edinburgh Napier University. He received his PhD in computer science from the University of Technology of Compiegne, France in 2005. Before joining Edinburgh Napier he was a network RD engineer at Motorola Labs in Paris. Dr Romdhani is an active member within the Internet Engineering Task Force (IETF). He contributed to form the MultiMob (Multicast Mobility) working group to standardize IP mobile multicast communication protocols and reviewed many Internet drafts and standards including RFC 5757, RFC6224 and RFC6636.

1 Introduction

Internet of Things (IoT) is one of the main communication development in the last decade. According to [AIM10], the basic concept behind the IoT is the pervasive presence of various wireless technologies such as Radio-Frequency IDentification (RFID) tags, sensors, actuators or mobile phones in which computing and communication systems are seamlessly embedded. Through unique adressing schemes, these objects interact with each other and cooperate to achieve common tasks.

Technology advances along with increasing demand will foster a wide spread deployment of IoT's services, which would radically transform our corporations, communities and personal spheres. From the perspective of a private user, IoT's introduction will play a leading role in several services. E-health is seens as one of the most interesting applications as it will provide medical monitoring to millions of elderly and disabled patients while preserving their autonomy and comfort. By using body sensors, physiological data is gathered and transmitted to qualified medical staff that can intervene in case of emergency. Nevertheless, e-health applications are unlikely to fulfil a widespread deployment until they provide strong security foundations. Securing communications in e-health applications necessarily passes through key management protocols that distribute security credentials between involved entities. However, the lack of energy power and computational capabilities in such kind of environment hinder the deployment of classic developed security solutions.

MIKEY-Ticket [MT11] is a key management protocol characterized by its simplicity and adaptation to centralized architectures. In fact, these architectures are interesting to be considered for resource constrained environments, as there is no need to pre-distribute credentials. By using these kind of architectures, users can request security credentials only when required. Centralized solutions also scale well when the number of users grows. Additionally, MIKEY-Ticket specifies different message exchanges that can be transported over UDP and integrated within several security protocols (e.g. IPSEC, DTLS, HIP).

MIKEY-Ticket needs to be tailored for constrained environments in order to adapt to resources constraints of such environments. To this end, we introduce two solutions to tailor MIKEY-Ticket to e-health environments without weakening its security properties. In the first solution, we propose a new 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks) header compression scheme for MIKEY-Ticket. Our scheme is intended to save energy and avoid 6LoWPAN fragmentation that may occur when a datagram size exceeds the link layer MTU (Maximum Transmission Unit of the IEEE 802.15.4 protocol). Indeed, fragmentation is undesirable, as 6LoWPAN is vulnerable to fragmentation attacks [HHW+13]. In the second solution, we propose a new exchange mode to reduce the number of exchanged messages from six to four. The main concern being to reduce the involvement of the constrained nodes in the exchange process.

To assess our proposed adjusted MIKEY-Ticket protocol with respect to its security properties and energy savings, we have proceeded with a theoretical analysis that we have further formally validated through an implementation in Avispa tool [refa]. In addition, based on energy models that consider both communication and computational costs, we have estimated the energy savings at the constrained nodes side. Our results show a progressive gain of energy cost according to the compression rate level while preserving the MIKEY-Ticket security properties.

The rest of this paper is organized as follows. In Section 2, e-health applications in the context of IoT are briefly introduced along with the main security threats that might limit their deployment. Thereafter, we provide in Section 3 an overview on the state of the art of the proposed security approaches. In Section 4, we introduce the motivations behind our choice of MIKEY-Ticket over other existing protocols. Furthermore, for a proper understanding of our contribution, we also present the different technologies used. We outline our network scenario in Section 5. In Section 6, we describe in detail how we have adjusted MIKEY-Ticket. Both security and quantitative analysis of our contribution are provided in Section 7. Finally, Section 8 concludes the paper and gives future directions.

2 E-health applications in the context of Internet of Things

Internet of Things deployment will open doors to a huge number of applications that would deeply improve our daily life. E-health applications are one of the typical applications that are gaining more and more attention [AIM10]. An e-health system is defined as a radio-frequency-based wireless networking technology that provides ubiquitous networking functionalities. It is based on the interconnection of tiny nodes enhanced with sensing and/or actuating capabilities planted, or placed around the human body. E-health applications are context-aware, personal, dynamic and anticipative by nature. As IoT is designed to meet these key characteristics, it provides a natural and suitable environment for their efficient deployement. In fact, an extensive research study on using IoT paradigm in e-health has recently been reported [IJSP10]. Population ageing and the increase of survival chances from disabling accidents and illnesses will lead to an increased demand from today's population that requires

a continuous health care and monitoring [DMOD$^+$].

E-health applications could spare a patient from being admitted in hospitals for a long period of time. Reducing the number of nights that a patient may spend in a hospital and the associated risks that may result is a key area of focus for the medical community. Additionally, a continuous monitoring capability, if available, can anticipate the need for an emergency intervention. Moreover, early stage diagnostics could also be achieved remotely [PW10]. In brief, e-health applications in the context of IoT constitute a cost effective and unobtrusive solution that is of best interest of todays patients. Nevertheless, e-health applications are seriously challenged by many security threats that limit their large scale deployment.

Studies in [LL10][JR13][LOCL10][NST06] have underlined that e-health applications might be more vulnerable to attacks compared to other IoT applications as the generated data is highly sensitive and private. The health related records are always private in nature, and any security breach in the confidentiality of such data would seriously repulse patients from adopting e-health solutions. For instance, many people would not like their personal health information, such as early stage of pregnancy or details of certain medical conditions, be divulged to third parties [ALK12]. In fact, the eavesdropped communications could be used for several illegal purposes. Moreover, any eventual modification of health related captured data could lead to disastrous consequences as it could engender wrong medical prescription or delay an emergency intervention.

Classical countermeasures are not suitable to the constrained environment of IoT due to several factors such as power and computation limitations, weak reliability of wireless links and the scalability issue. Thus, a considerable effort has been made by the research community to provide viable solutions to secure IoT applications. The next section provides an in-depth overview on the state of the art of the proposed security approaches and explains the motivations behind our contribution.

3 Related work

The research community attempted to propose security protocols that take into consideration the constrained resources of IoT. In this context, we distinguish two distinct research directions: i) specific solutions for e-health applications, and ii) the tailoring of standard security protocols for the IP-based IoT.

Several *specific solutions for e-health applications* have been proposed in the literature. For instance, hardware solutions are proposed to deal with the scarcity of resources [HNL08] [MRL06]. However, these approaches still present some drawbacks as they do not offer AES (Advanced Encryption Standard) decryption (only base stations can decrypt the transmitted data). In addition, they are highly platform-dependant and not all the nodes are equipped with hardware encryption capabilities. Besides, TinySec is part of the official TinyOS release that aims to achieve link-layer encryption and authentication of data in biomedical sensors [KSW04]. This protocol is based on a single key shared among nodes which constitutes its main weakness as node capture would give access to the entire network. A different approach based on biometric techniques is therefore proposed [CVG03] [PZB06]. These techniques use the human body to manage the key establishment process based on physiological values (e.g., electrocardiogram).

A different but complementary research direction has seen several interesting approaches that aim to *tailor security protocols for the IP-based IoT*. The main focus of these works is to make standard based security protocols suitable for constrained IoT environments. In particular, several compression schemes for the IP-based IoT have been proposed. The compression of IPv6 headers, extension headers along with UDP (User Datagram Protocol) headers has been standardized through the 6LoWPAN adaptation layer in [MKHC07] [HT11]. Moreover, authors in [GMS10] and [RDC$^+$11] have presented 6LoWPAN based compression techniques for IPsec payload headers: AH (Authentication Header) and ESP (Encapsulating Security Payload), that have been later standardized in [RDS13]. Besides, an IKE (Internet Key Exchange) compression scheme has been also proposed in order to provide a lightweight automatic way to establish security associations for IPsec [RVJ12]. Likewise, header compression layers for DTLS (Datagram Transport Layer Security) and HIP DEX (Host Identity Protocol Diet Exchange) were respectively introduced in [RTV12] and [HHHW13].

Apart from packet compression schemes, several delegation procedures of protocol's primitives have been proposed to offload the computational load to third entities. Authors in [SO12a], [SO12c] and [SO12b] have introduced collaboration for HIP (Host Identity Protocol). The idea is to take advantage of more powerful nodes in the neighborhood of a constrained node to carry heavy computations in a distributed way. Likewise, IKE session establishment delegation to the gateway have been proposed in [BBL$^+$12]. Furthermore, authors in [FHM$^+$07] have introduced a delegation procedure that enables a client to delegate the certificate validation process to a third party. While delegation approaches reduce the computational load at the constrained node, they introduce the use of a trusted third party. As a result, the end to end property is no longer ensured with respect to protocols, which initially were designed to ensure it. Further design improvement approaches have been introduced to tailor end-to-end security

protocols to IoT. For example, authors in [HWZ+13] have proposed complementary lightweight extensions to HIP DEX that could be generalized to DTLS and IKE. Following the same way, authors in [HZS+13] have introduced design ideas to reduce the overhead of the DTLS handshake where, their primary goal was to make the use of certificates for authentication purposes viable in IoT contexts. Besides, authors in [BOO13], have proposed an approach to mitigate the DoS attack in MIKEY-Ticket.

We do believe that securing IoT applications will be achieved through tailoring current security protocols to IoT environments rather than developing new specific solutions. To the best of our knowledge, no prior solutions have been proposed to tailor the MIKEY-Ticket protocol to the constrained IoT environment.

4 Background

4.1 MIKEY-Ticket choice

In this subsection, we focus on the motivations that are behind our choice of MIKEY-Ticket over other existing protocols, particularly IKE. This latter is a key exchange protocol that aims to perform mutual authentication and to provide Security Associations (SAs) to be used as input for IPsec. Indeed, securing IoT communications at the IP level is likely to be achieved through the use of IPsec [KKT14]. In this way, MIKEY-Ticket aims to achieve the same goal. In our study, we have focused on MIKEY-Ticket instead of the widely adopted IKE. The reasons behind this choice are the following.

- The proposed e-health constrained scenario involves the use of tiny nodes that are highly limited by their computational capabilities. In fact, during its first request/response exchange process (i.e.IKE_SA_INIT), IKE involves the two parties in a Diffie-Hellman instantiation phase, which requires an important energy consumption due to exponential operations. Indeed, Public key operations are not suitable for highly constrained environments. Besides, the Pre-Shared mode of MIKEY-Ticket only involves symmetric operations, which are much more energy saving compared to asymmetric approaches [WGE+05]. Furthermore, Mikey- Ticket is designed to involve a central trusted entity which makes it more suitable to our network scenario compared to IKE. The trusted entity has a double role to play. Firstly, it acts as a gateway (i.e. 6LoWPAN Border Router) through which 6loWPAN headers are compressed and decompressed. Secondly, it spares the constrained node from using public key cryptography by generating and distributing the required security credentials.
- MIKEY-Ticket is a product of the IETF (Internet Engineering Task Force) such as IKE [KHNE10], DTLS[ER11] and other standard based protocols. In fact, our approach to address data confidentiality in IoTs applications aims to propose new extensions to standardized protocols in order to adapt them to the IoT context. Following this approach, MIKEY-Ticket sounds to be the adequate protocol that can be extended to ensure secure communications in IoT.
- A lot of efforts have been carried out by the research community to optimize the IKE protocol. As IoT is only in its first stages of deployment, the protocol suite that should be implemented to secure IoT based applications is not clear yet. Our research effort attempts, therefore, to bring a contribution in this process of adapting and selecting existing protocols for IoT environments.

4.2 MIKEY-Ticket overview

MIKEY-Ticket [MT11] is a key distribution protocol designed to enhance the Multimedia Internet KEYing protocol (Mikey) [ALNN04]. It defines new modes of key distribution which are well adapted to centralized based scenarios where a third trusted entity is available. MIKEY-Ticket considers two entities that aim to establish a shared secret. One of the two entities assumes the Initiator role whereas the second one assumes the Responder role. The key establishment relies on a Key Management Server to generate and deliver the needed credentials. Such design spares the peers from a pre-distribution phase that would require credentials storing. Instead, peers can request such credentials only when required. In this work, we only consider the Pre-Shared Key mode (PSK) of MIKEY-Ticket as the Public Key (PK) mode and the Diffie-Hellman key exchange mode are ruled out due to their inadequacy with IoT constrained environments.

We provide a brief description of MIKEY-Ticket message exchanges and the general MIKEY header (HDR) format. Table 1 summarizes the used notations.

4.2.1 Message exchanges

MIKEY-Ticket uses six messages to establish a new key between the Initiator I and the Responder R (see Figure 1). The protocol relies on the Key Management Server (KMS) which delivers the generated key. The Initiator and the Responder do not share any credentials. Instead, they share a secret master key with the *KMS*. This key is used to derive an authentication key and an encryption key. The generated keys are used to secure the communication for I and R whereas *KMS* provides data authenticity, data integrity and confidentiality.

We briefly describe the content of each exchanged message of the full three round-trip MIKEY-Ticket

Table 1 Terminology Table

Notation	Description
I	Initiator
R	Responder
KMS	Key Management Server
X_{ID}	The Identity of X
N_X	Nonce generated by X
$K_{X,Y}$	Shared key between X and Y
$aK_{X,Y}$	Shared authentication key between X and Y
$eK_{X,Y}$	Shared encryption key between X and Y
$[data]_K$	Data encrypted with the key K
$Ticket$	Object used to identify and deliver keys

mode:

REQUEST_INIT: through this message, node I expresses its willingness to establish a shared key with node R. The message contains information about the responder's identity. To ensure authenticity, a Message Authentication Code (MAC), which is computed with $aK_{I,KMS}$, is included.

REQUEST_RESP: after successful verification, the request is authorized and the *KMS* generates the requested key K and encodes it in a ticket. The message is sent to I.

TRANSFER_INIT: upon reception of REQUEST_RESP message, node I derives an authentication key aK and an encryption key eK to secure data transmission between I and R. Then, node I transfers the ticket to R through TRANSFER_INIT message. Also, a MAC is computed using aK and included in the message.

RESOLVE_INIT: through this message, node R asks the *KMS* to return the key K encoded in the ticket. The message is protected by a MAC based on $aK_{KMS,R}$.

RESOLVE_RESP: if node R is authorized to receive the generated key encoded in the ticket, the *KMS* sends RESOLVE_RESP message that includes the generated key K. The message is protected through encryption and a MAC message based on $aK_{KMS,R}$.

TRANSFER_RESP: R is in possession of the generated key K. TRANSFER_INIT's MAC can thus be checked. The exchange is concluded through TRANSFER_RESP message to prove the correct reception and derivation of the generated session key.
It is worth noticing that the different messages contain a nonce for protection against replay attacks.

Figure 1 depicts the signaling for the full three round-trip MIKEY-Ticket mode. Nevertheless, RFC 6043 [MT11] introduces four different modes according to the specificities of both the Initiator and the Responder. Mode 1 represents actually the full three round-trip mode where only the *KMS* is in charge of generating, deriving and distributing the keying materials. Both I and R have to request/resolve messages with the *KMS*. In mode 2, the exchanges between the *KMS* and R are omitted (i.e. *RESOLVE_INIT* and *RESOLVE_RESP*). However, R has to be able to resolve the ticket without assistance from the *KMS*. In mode 3, the *ticket request* exchange (i.e. *REQUEST_INIT* and *REQUEST_RESP*) can be omitted if I is able to create the keying materials without an assistance from *KMS*. Mode 4 only contains a *ticket transfer* exchange (i.e. *TRANSFER_INIT*). However, it requires from I and R to share security credentials prior to the start of the procotol session.

4.2.2 Common Header Format (HDR)

The Common Header payload (see Figure 2) contains information about the different exchanged messages. It is always present as the first payload in each message. In the following, we present a succinct description of each field contained in the Mikey_Ticket header. We refer to RFC3038 [ALNN04] and RFC6043 [MT11] for a more detailed description:

- *Version (8 bits):* version of Mikey.
- *Data type (8 bits):* type of the exchanged message.
- *Next Payload (8 bits):* identifies the payload added after the current payload.
- *V (1 bit):* flag to indicate the use of a verification message.
- *PRF func (7 bits):* indicates the key derivation function.
- *CSB ID (32 bits):* Crypto Session Bundle (CSB) is a collection of one or more Crypto Sessions (CS). CSB ID field identifies the CSB.
- ♯ *CS (8 bits):* a Crypto Session refers to a data steam protected by a single instance of a security

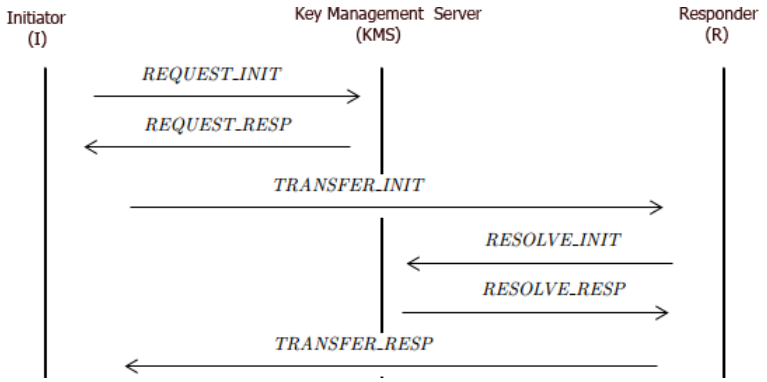

Figure 1: MIKEY-Ticket full three round-trip mode exchange (RFC 6043)

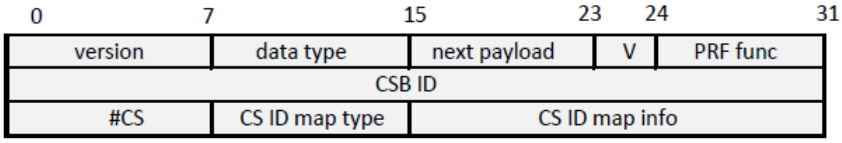

Figure 2: MIKEY Common Header Format (RFC 3830)

protocol. ♯ CS field indicates the number of Crypto Sessions within the CBS.

- *CS ID map type (8 bits):* specifies the method of uniquely mapping crypto sessions to the security protocol sessions.
- *CS ID map info (variable length)* identifies and maps crypto sessions to the security protocol sessions.

4.3 6LoWPAN Adaptation Layer

The 6LoWPAN standard defined in [HT11] aims to transfer IPv6 packets to IEEE 802.15.4 based networks. 6LoWPAN uses IPV6 header compression mechanisms of IPv6 datagrams. Compression mechanisms are motivated by the limited space available in 802.15.4 frames to encapsulate IPv6 packets. In fact, the size of the 802.15.4 frame payload (102 bytes) leaves limited space for an IPv6 packet as 48 bytes are required only for its header. 6LoWPAN defines encoding formats for compression based on shared state within contexts. In other words, it takes advantage of the fields that are implicitly known to all nodes in the network or can be deduced from the MAC layer. The compression scheme consists of IP Header Compression (IPHC) and Next Header Compression (NHC).

IPHC encoding describes how an IPv6 header is compressed. As depicted in Figure 3, 13 bits of the 2 bytes long IPHC are used for compression. The IPv6 header fields that are not compressed are placed immediately after IPHC. Moreover, NH field in IPHC indicates whether the following header is encoded using NHC. If so, NHC encoding follows immediately the compressed IPv6 header. Compression formats for different next headers are identified by a variable ID bits plus the specific header compression encoding bits. The NHC to encode IPv6 extension headers and UDP header are already defined. For more details on 6LoWPAN, we refer the reader to RFC 6282 [HT11].

5 Network scenario

We consider a scenario of an e-health application where smart objects (contextual sensors), gateways and remote entities are used (see Figure 4). IP-enabled smart objects are in charge of sensing health related data (e.g. blood pressure, blood glucose level, temperature level, etc.). They are planted in the human body. Gateways connect these objects to a backend infrastructure such as Internet. It is worth mentioning that user's smartphones could be used as gateways. Remote entities are in charge of processing and analyzing the received data.

Smart objects have limited computational power, memory and energy resources, whereas gateways are much less resource constrained and are comparable to standard routers. Remote entities can take the form of a server hardware or being distributed in a Cloud infrastructure with dynamic resources.

The mapping with MIKEY-Ticket concepts is defined as follows:

- *Initiator*: smart object (e.g. IP-enabled tiny sensor).

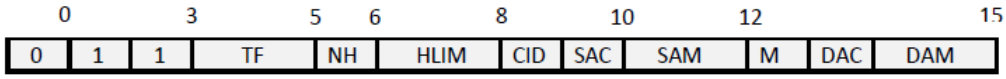

Figure 3: IPHC

- *Key Management Server*: gateway (e.g. smartphone).
- *Responder* : remote entity (e.g. servers disposed in hospitals).

Securing e-health applications relies on efficient key management schemes that ensure reliable key distribution. We do believe that the best approach to tackle security challenges in the evolving IoT is to focus our efforts on standard based protocols. We have chosen MIKEY-Ticket for its simplicity and its adaptation to centralized scenarios which suits well our e-health application. However, current key management protocols such as MIKEY-Ticket were designed to be used in an unconstrained environment which does not take into consideration resources limitation. In the next section, we present in detail our contribution to make MIKEY-Ticket more lightweight while preserving its security properties.

6 Reducing the overhead of MIKEY-Ticket

In order to reduce the communication overhead of MIKEY-Ticket protocol when implemented on constrained entities, we have adopted two complementary approaches. Firstly, we have reduced the size of the exchanged messages by proposing a new header compression scheme. Secondly, we have minimized the number of exchanged control messages by proposing a new exchange mode.

6.1 New header compression scheme

In this section, we describe our proposed 6LoWPAN header compression scheme for MIKEY-Ticket. Our compression is based on the fact that the fields which are implicitly known to all entities in the network or those that can be deduced from the MAC layer can be removed. As explained in section 4.2, the NHC is used to encode the IPv6 extension headers and UDP header. Nevertheless, despite 6LoWPAN has defined header compression for UDP, no NHC compression is defined in case where headers contained in UDP payloads are compressed. In fact, MIKEY-Ticket common header is contained in the UDP payload. Therefore, we propose to use the 6LoWPAN extension proposed in [RTV12] to extend 6LoWPAN header compression mechanisms. These extensions indicate that the headers of protocols that are part of the UDP payload are compressed with 6LoWPAN-NHC.

MIKEY-Ticket common header is 12 bytes long. It is appended to each packet through the different exchanged messages. We propose a 6LoWPAN-NHC to compress MIKEY-Ticket header called 6LoWPAN-NHC-HDR. The proposed approach allows to reduce the header length from 12 bytes to 3 bytes (2 bytes for our 6LoWPAN-NHC-HDR plus 1 byte for the Next Payload field that is always carried inline) in the best compression case. In fact, only 13 bits are required to encode the different fields. Nevertheless, in order to remain standard compliant (i.e. the size of NHC encodings is multiple of bytes), our 6LoWPAN-NHC-HDR is 2 bytes long. 6LoWPAN-NHC encoding schemes do not limit the length or the value of the NHC-ID. However, the NHC-ID must be unique in order to distinguish the various existing compressed 6LoWPAN headers (e.g. IPv6, UDP, IPsec, DTLS, IKE). Thus, the first four bits implement the ID field to uniquely identify our NHC encoding. We set the ID bits to 1100. To the best of our knowledge, the 1100 bits are currently unused as NHC identifiers. In the following, we present in detail the encoding approach for each field (see Table 2 and Figure 5).

- *Version (V):* if 0, the version is the default and latest MIKEY-Ticket version defined in [ALNN04] and the field is skipped. If future versions are defined, the bit is set to 1 and the version number is carried inline after the 6LoWPAN-NHC-HDR header. Our compression is thus kept dynamic and flexible.

- *Data type (DT):* the data type field describes the type of the exchanged messages. Based on our new exchange mode (See section 4.1), we only consider three types of messages (i.e. REQUEST_INIT, REQUEST_RESPONSE, TRANSFER_END) plus the ERROR type. In addition, only the Pre-Shared Key (PSK) mode is considered. The other modes are ruled out due to their inadequacy with our constrained network scenario. Doing so, we are then able to use just 2 bits encoding for the data type field instead of 8 bits in the original version:

  00: REQUEST_INIT
  01: REQUEST_RESPONSE
  10: TRANSFER_END
  11: ERROR

- *Verification V (VF):* the VF field encoding is similar to the non-compressed header. If it is set to 0, no verification message is used. When it is

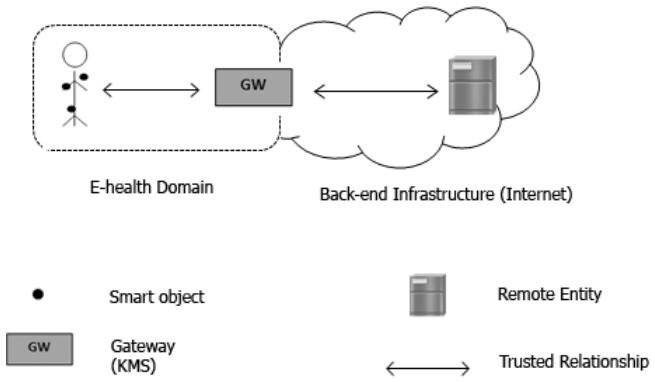

Figure 4: Network Scenario

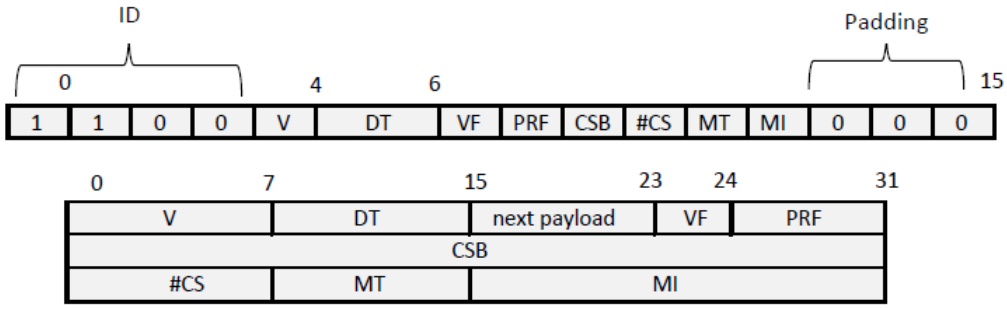

Figure 5: Our 6LoWPAN-NHC-HDR encoding compared to the basic MIKEY's header

set to 1, a verification message is required.

- *PRF func (PRF):* if 0, the default PRF function defined in [ALNN04] is used. If set to 1, the PRF function value is carried inline.

- *CSB ID (CSB):* the CSB ID is chosen by the Initiator and needs to be unique between each Initiator-Responder pair. Instead of carrying its 32 bits size inline, we propose to derivate the CSB ID from the concatenation of lower layer addresses. To guarantee uniqueness, we require the use of unique identifiers such as 6LoWPAN addresses, or physical addresses. One bit is sufficient for the encoding. If set to 0, the CSB ID is derived instead of being carried inline. If set to 1, the 32 bits CSB ID are carried after the 6LoWPAN-NHC-HDR header.

- ♯ *CS:* if we assume in our constrained scenario that there is only one CS in each CSB, there is no need therefore for keeping 8 bits to indicate the number of crypto sessions. We are then able to encode the ♯ CS with 1 bit. If this bit is set to 0, only one CS is considered. In addition, to make our compression flexible, if the bit is set to 1, the number of CS is carried inline.

- *CS ID map type(MT):* if 0, the default GENERIC-ID map type defined in [MT11] is used. If set to 1, the CS ID map type is carried inline.

- *CS ID map info (MI):* the CS ID map info size is kept variable in [MT11]. If we assume that there is only one CS in each CSB, we could use 1 bit for the encoding. If 0, the unique CS is identified with its corresponding mapping to the security protocol for which security associations are created. If set to 1, the map info field is carried inline.

The next payload field is always carried inline as it is impossible to predict or deduce the next payload content. In addition, the three last bits are used as padding bits to remain standard compliant with RFC6282 [HT11] (NHC size is defined as 2 bytes long).

6.2 New MIKEY-Ticket exchange mode

Our new communication exchange mode for MIKEY-Ticket is designed to minimize the involvement of constrained nodes. We consider the constrained node as the Initiator of the protocol and the remote entity as the Responder. The constrained node is in charge

Table 2 MIKEY-Ticket Common Header compression

Field (sizes in bits)	MIKEY Common Header	Our 6LoWPAN-NHC-HDR
Version (V)	8	1
Data type (DT)	8	2
Next Payload	8	8
Verification V (VF)	1	1
PRF func (PRF)	7	1
CSB ID (CSB)	32	1
♯ CS	8	1
CS ID map type (MT)	8	1
CS ID map info (MI)	Variable length	1

of requesting the establishment of a session key with the remote entity and periodically sending updates. We assume that I and R are sharing security credentials with the *KMS* that is in charge of generating, deriving and delivering the required keying materials. Besides, AES-CTR (AES in Counter Mode) algorithm, which is specified as mandatory-to-implement in RFC 3830 [ALNN04] is used for encryption. Also, AES-CBC (AES in Cipher Block Chaining mode) is used for MAC computation. Our communication exchange mode is depicted in Figure 6 and Table 1 summarizes the different notations used. It is worth mentioning here that although mode 2, mode 3 and mode 4 (see section 4.2.1) introduced in RFC 6043 [MT11] reduce the number of exchanged messages compared to the full three round-trip mode, they introduce strong assumptions on the ability of both I and R to either handle the generation and distribution of security credentials or to share credentials prior to the start of the session. For these reasons, our proposed exchange mode can be considered as an extension of the proposed exchange modes defined in RFC 6043 [MT11]. In fact, our new exchange mode does not assume any capabilities regarding neither I nor R as it is intended to be adaptable to constrained e-health scenarios.

REQUEST_INIT: the Initiator starts the exchange process by sending a *REQUEST_INIT* message to *KMS*. This message contains the identities of I (I_{ID}), *KMS* (KMS_{ID}), and R (R_{ID}). In addition, it contains a nonce N_I generated by I, which will be used as a session identifier. Furthermore, node I computes a *MAC* using $aK_{I,KMS}$ to ensure message authenticity. The message is then sent to *KMS*. *REQUEST_INIT* has the following structure: $\{[I_{ID}, R_{ID}, KMS_{ID}, N_I]_{eK_{I,KMS}}, MAC\}$.

REQUEST_RESPONSE: when *KMS* receives the *REQUEST_INIT* message, it validates the *MAC* using $aK_{I,KMS}$. Upon successful verifications, *KMS* decrypts the message using $eK_{I,KMS}$ and retrieves the different identities and the nonce N_I. If the request is authorized, *KMS* generates the requested key $K_{I,R}$ and uses the key derivation function defined in RFC3830 [ALNN04] to derive both $aK_{I,R}$ and $eK_{I,R}$. Then, *KMS* constructs two versions of *REQUEST_RESPONSE* message. The first message is intended to I. It is encrypted with $eK_{I,KMS}$ and contains a *MAC* computed using $aK_{I,KMS}$. In addition, the message contains the nonce N_I. The second message is intended to R. It contains a *MAC* computed using $aK_{KMS,R}$ and is encrypted using $eK_{KMS,R}$. In addition, *KMS* generates a nonce N_{KMS} and includes it in the message along with N_I. The *REQUEST_RESPONSE* is intended to node I, and has the following structure: $\{[I_{ID}, R_{ID}, KMS_{ID}, aK_{I,R}, eK_{I,R}, N_I]_{eK_{KMS,I}}, MAC\}$. The *REQUEST_RESPONSE* intended to R has the following structure: $\{[I_{ID}, R_{ID}, KMS_{ID}, aK_{I,R}, eK_{I,R}, N_I, N_{KMS}]_{eK_{KMS,R}}, MAC\}$. The two versions are then sent to I and R.

TRANSFER_END: upon receiving a *REQUEST_RESPONSE* message, R checks the freshness of N_{KMS} and validates the *MAC* using $aK_{KMS,R}$. Upon successful verification, R decrypts the message and retrieves both $aK_{I,R}$ and $eK_{I,R}$. Node I proceeds similarly and retrieves $aK_{I,R}$ and $eK_{I,R}$ upon receiving *REQUEST_RESPONSE* message. R constructs *TRANSFER_END* as a verification message. It includes the nonce N_I and computes a *MAC* using $aK_{I,R}$. The message is then sent to I. This message has the following structure: $\{[I_{ID}, R_{ID}, N_I]_{eK_{I,R}}, MAC\}$. Upon receiving *TRANSFER_END* message, node I checks the freshness of N_I to avoid any replay attack and validates the *MAC*. A successful verification is considered as a proof of R's knowledge of both $aK_{I,R}$ and $eK_{I,R}$.

Our new communication exchange mode reduces therefore the number of exchanged messages from six to four messages compared to the basic MIKEY-Ticket defined in RFC 6043 [MT11] regardless of the ability of I and R to generate, derive or distribute security credentials. As a result, the constrained node processes and exchanges fewer messages.

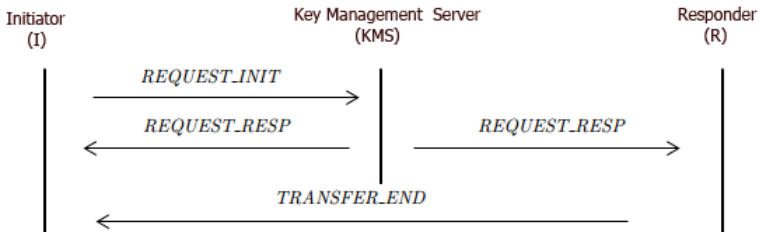

Figure 6: New MIKEY-Ticket exchange mode

7 Analysis

In this section, we provide a detailed analysis of our proposed tailoring for MIKEY-Ticket both in terms of security analysis and energy consumption. Firstly, we conduct a theoretical security analysis of our new exchange mode. In addition, we analyze our protocol's behaviour against the well-known attacks that could hinder the establishment of a secure channel in an e-health environment. Our analysis is then validated using an automated validation tool called Avispa [refa] which is based on formal models. After validating the security properties, we focus on the energy gain of our approach. Different energy models are used to estimate the total energy cost composed of both computational and communication costs. The results are compared with the basic version of MIKEY-Ticket.

7.1 Security analysis

7.1.1 Key exchange properties

The security features of our new MIKEY-Ticket exchange mode have been assessed based on the properties presented in [RALS11]. We have added extra analysis concerning integrity and confidentiality as we consider them critical for e-health applications. Hereafter, our communication channel is split into two parts or segments: Seg1) from the Initiator to the *KMS* and Seg2) from the *KMS* to the Responder.

- *Confidentiality:* The exchanged data between the different entities involved in our protocol are kept confidential. According to [ALNN04], AES-CTR is the default and mandatory-to-implement encryption algorithm. Nowadays, more and more tiny sensors include AES hardware coprocessors which help to decrease the overhead. For *Seg1*, encryption is based on the encryption key $eK_{I,KMS}$ shared between the Initiator (constrained node) and the *KMS*, whereas in *Seg2*, encryption is ensured by the use of the encryption key $eK_{KMS,R}$ shared between the *KMS* and the Responder (remote server). In addition, periodical updates of the established keys are required in order to strengthen the confidentiality and prevent long term attacks.

- *Authentication and integrity:* By using *MAC* messages either in *Seg1* or in *Seg2* communication parts, our new exchange mode ensures that the exchanged data is genuine. In particular, it ensures that data has not been altered and has been sent from legitimate nodes. *MAC* messages are computed and appended to the exchanged messages based on AES-CBC mode using $aK_{I,KMS}$ in *Seg1* and $aK_{R,KMS}$ in *Seg2*. Furthermore, nonces (e.g. time-stamps or random values) are included in the exchanged messages to avoid replay attacks.

- *Distribution:* The distribution of security credentials in both communication segments is performed by an offline dealer during the initialization phase. This constitutes one of the major drawbacks of key distribution schemes based on a pre-shared context. In return, these schemes simplify the cryptographic operations (i.e. Symmetric) at the nodes side which is highly desirable in constrained environments. Besides, upon the establishment of a shared context, our new exchange mode can be run in an online manner, which allows autonomous update procesing.

- *Overhead:* The computation overhead is particularly low. Our compression scheme allows a considerable improvement in energy consumption as the size of the exchanged messages is reduced. Moreover, the constrained nodes are involved in fewer messages compared to the full three round-trip MIKEY-Ticket mode (see Figure 1 and Figure 6). Constrained nodes are thus less solicited as they take advantage of the shared pre-established context with the *KMS*. A more detailed analysis regarding energy consumption is provided in section 7.2.

- *Resilience:* The resilience of our scheme is high. In fact, the loss of a node and thus its key affects only the corresponding sensor as each sensor only

stores its shared key with the *KMS* (i.e. $K_{R,KMS}$) and eventually an established key with I (i.e. $K_{I,R}$). The *KMS* maintains a different key with each constrained node either for the pre-shared context or for the generated shared key.

- *Extensibility and scalability:* Our network model allows new sensors as well as new remote entities to be added (e.g. we can imagine a physician prescribing the implantation of a new sensor for medical reasons). An offline dealer will have to establish a shared context between the new entities and the *KMS*. No extra operation is required from existing constrained nodes or remote entities when new nodes join them. As a result, high scalability is ensured which is particularly required for constrained environments.

- *Storage:* Smart objects now provide considerable amounts of storage space due to recent advances in flash memory technology [TD11]. Moreover, our new exchange mode does not add further credentials to be stored in the constrained nodes. The amount of data to be stored is limited, as only two keys (i.e. $K_{R,KMS}$ and $K_{I,R}$) have to be stored. Storage space will therefore not limit the deployment of our scheme.

7.1.2 Protocol behaviour against e-health well-known attacks

E-health applications are subject to several attacks that threaten the establishment of secure channels [LL10] [JR13] [LOCL10]. In this section, we analyze the behaviour of our protocol against these attacks. We focus on the attacks that occur in the network and transport layers of the OSI (Open System Interconnection) model.

Ensuring key freshness is an important concern with regards to our new MIKEY-Ticket exchange mode. Indeed, to provide the perfect forward secrecy property, the involved entities have to be able to detect replayed messages. In particular, e-health applications might be more vulnerable compared to other types of applications as an outdated information could lead to inadequate and serious medical consequences. To overcome this issue, we have introduced the use of nonces in the different exchanged messages. In fact, these nonces are implemented using one of the following strategies according to the network segment, and to the constrained node capabilities:

- Random numbers
- Sequence numbers
- Timestamps

Random numbers might constitute a solution in our e-health scenario. The constrained node (i.e. the Initiator) maintains a list of the previous received random values in its internal memory. Upon receiving a new message, the initiator checks if the nonce has already been received. As a result, replayed messages are detected. This solution brings a drawback ; the constrained node has to maintain a list of the received nonces in its internal memory. This issue can be attenuated by the storage capacity of new developed nodes [TD11]. The second solution is based on sequence numbers, which does not require any data storage. Sequence numbers provide a sequential counter in the exchanged messages. In case where a message is replayed, its counter will be smaller or equal to the current one. Thus, the message will be dropped. However, if the KMS goes down (e.g. reboot, hardware failure, etc.), this protection is no longer effective. In fact, the KMS will lose track of the current counter value. Besides, to ensure message freshness, timestamps could also be used. This solution is not suitable for constrained devices as it consumes a lot of energy. In fact, synchronized clocks have to be maintained between the KMS, the remote server, and the constrained nodes.

Taking into account our network specifications, we discuss the feasibility of the precedent solutions. It is obvious that maintaining clock synchronization between KMS and the constrained nodes is not feasible. However, this solution is adopted to protect the unconstrained part of the network model, namely the channel linking the KMS with the remote server (Seg2). In fact, the Responder and the KMS are not able to challenge each other and they are considered as non-constrained entities that are able to maintain clock synchronization between them. Hence, the nonces are implemented as timestamps. By doing so, the KMS and the remote server will easily prevent replay attacks.

Regarding Seg1 communication part, our proposed exchange mode allows the Initiator to challenge the KMS about the nonce. In addition, the constrained node is not able to maintain clock synchronization with the KMS. Consequently, the solution based on random numbers (or sequence numbers) is adopted. If the storage capacity of smart objects is very limited, the solution based on sequence numbers is prefered at the expense of ensuring a highly reliable entities with small probabilities of failure. If storage capacity is not a concern, the solution based on random numbers can be adopted. In brief, protecting our new exchange mode against replayed messages is achieved through the combination of the above discussed strategies according to the network model specificities.

Denial of Service (DoS) attacks could seriously threaten the availability of our e-health application. In fact, the gathered health related data should always be available even if the system is under a DoS attack.

Like the basic version of MIKEY-Ticket, our new exchange mode is protected against DoS attacks by using the same techniques. In particular, the KMS does not establish any internal state before authenticating both the remote server and the constrained nodes. The different parties share a long-term key with the KMS. Each exchanged message is authenticated before being processed. Besides, classical countermeasures such as rate-limiting and ACL (Access Control List) could also be implemented. Any malicious message would lead to an abortion of the protocol execution. Node redundancy could be another option. Whenever an entity is made unavailable due to a DoS attack, the protocol execution carries on with the redundant backup node. We refer to [ALNN04] and [MT11] for a more detailed analysis of MIKEY-Ticket behaviour regarding DoS attacks.

Sybil attacks [Dou02] [JR13] where a node claims multiple fake identities could lead to harmful consequences in the context of e-health applications. Using these attacks, an intruder could use feigned identities to send false information. As a result, either genuine emergency situations are skipped, or ceaseless false emergency situations are thrown. Our protocol is protected against Sybil attacks. There is no way for a malicious node to perform a Sybil attack, unless the KMS (assumed to be a trusted entity) is corrupted. In fact, long term keys are shared between the KMS, the Initiator (i.e. sensor), and the Responder (i.e. remote server). Any exchanged message with the KMS contains the identity of the sender, and is authenticated using the pre-shared long term keys. In addition, before any further processing, the KMS checks its access control policy regarding the sender.

Another point of interest regarding the threat model in e-health applications is the attacks that aim to drain the energy power of sensors, and therefore make them unavailable or force them to enter a sleep mode. For instance, the De-synchronization attack targets the sequence number of the exchanged messages. Actually, this will lead to infinite retransmissions which waste both energy and bandwidth resources. Providing message integrity is the main security concern that hinders this type of attacks. In fact, MAC messages are computed and checked for each exchanged message ensuring that the included data has not been altered.

E-health applications are subject to several routing attacks. Our key management protocol is not involved in securing the routing process, instead, it aims to establish a secure channel upon which the gathered data can be securely transmitted. In fact, we rely on other mechanisms regarding this aspect. Countermeasures usually involve the introduction of Intrusion Detection Systems (IDS) [RWV13] [LLL+12].

7.1.3 Formal validation

Several techniques have been introduced to model and formally validate a security protocol regarding its properties. Model checking [CGP99] is one of the formal methods used to validate finite-state-concurrent systems such as communication protocols. It usually involves verification tools to exhaustively search all possible execution sequences for desired properties in a protocol specification. Many security protocols have been validated through model checking [TCC+09] [HRZ08], and several validation tools are based on model checking [refa] [refc] [refb]. We highlight some advantages of model checking compared to classical approaches, which are developed around simulation, testing, and deductions:

- Gives the possibility to the users to check every single step of the execution process, allowing them to detect any malfunction in a highly accurate way. However, using simulation or testing, only a broad overview of the protocol behaviour is provided. In addition, some flaws might remain unfound until the protocol's production stage is initiated.
- Allows prompt and automated verifications through different tools that implement model checking. In fact, by adopting model checking, users can avoid prototyping their protocols.

AVISPA (Automated Validation of Internet Security Protocol and Applications) is a state-of-the-art verification tool for security protocols that includes a set of model checkers with a common front end. The tool follows the Dolev-Yao intruder model [DY81] to intercept messages or to insert modified data. It performs analytical rules to state whether the protocol is safe or not. In case of unsafety, the tool provides a trace highlighting the steps that led to the attack. In fact, Avispa is considered as an effective tool for the analysis of different Internet security protocols and applications. In the literature, several security protocols have been validated throught Avispa [CDS+11] [MC11] [CM09] [RMMLBLS06]. Moreover, the security protocols standardized by the Internet Engineering Task Force (IETF) have been analyzed by the AVISPA community (e.g. IKE, TLS, AAA), and some of the protocols have been found to be flawed [MD03] [refa].

The formal validation of our protocol was carried out using the same Avispa tool to prove that our new exchange mode does not violate the required security properties, in particular, confidentiality, authentication, delivery proof and replay protection. Protocol models in Avispa are written in a role-based language called High Level Protocol Specification Language, or HLPSL [CCC+04]. The actions of the different entities are specified in a module called *basic role*, while their interactions are defined by composing multiple *basic roles* together into a *composed role*. In addition, the

security goals of the analyzed protocol are specified in the *goal section* before launching the analysis. Besides, Avispa uses four different automatic protocol analysis techniques to validate the analyzed protocol against the specified security goals: on-the-fly model-checker (OFMC), constraint-logic based attack searcher (CL-AtSe), SAT-based model checker (SATMC), and tree automata based on automatic approximations for the analysis of security protocols (TA4SP).

In our modeling, we have first specified a *basic role* to describe the actions of the different entities involved. Then, we have specified how the participants interact with each other in a *composed role*. The security goals against which the protocol execution will be assessed have been specified in the *goal section*. Particularly, we modeled the confidentiality property of the generated $K_{I,R}$, in addition to the authentication property between the involved entities (i.e. I, KMS, and R).

For clarity reasons, we present our modeling using Alice-Bob ($A - B$) notation, where:

- *A: Constrained node*
- *B: Remote entity*
- *S: KMS*

The rest of the notations used are the same as those presented in Table 1.

- $A- > S : \{I_{ID}, R_{ID}, KMS_{ID}, N_I\}_{eK_{I,KMS}}, MAC$
- $S- > A : \{I_{ID}, R_{ID}, KMS_{ID}, aK_{I,R}, eK_{I,R}, N_I\}_{eK_{KMS,A}}, MAC$
- $S- > B : \{I_{ID}, R_{ID}, KMS_{ID}, aK_{I,R}, eK_{I,R}, N_I, N_{KMS}\}_{eK_{KMS,B}}, MAC$
- $B- > A : \{I_{ID}, R_{ID}, N_I\}_{eK_{I,R}}, MAC$

Upon completing modeling our exchange mode, we have checked its correctness using a protocol animation tool called SPAN [GG06] that has been introduced to help protocol developers in writing AVISPA specifications. The security goals were subsequently evaluated by executing the four Avispa's backends (i.e. OFMC, $CL - AtSe$, SATMC and TA4SP). Besides, we have used the default Dolev-Yao intruder model which allows to simulate an intruder that has full control over the network. All messages sent and received by the different entities might be intercepted, analyzed, modified (as far as the keys are known), or sent to other entities.

The results of the simulation were indicated in reports for each backend model produced by Avispa tool. Our new exchange mode is "SAFE" against OFMC (Figure 7), $CL - AtSe$ (Figure 8) and SATMC (Figure 9). However, against TA4SP database, the result was "INCONCLUSIVE". According to Avispa user manual [refa], an inconclusive result does not imply that an attack has been detected (Figure 10). Consequently, based on the obtained results, we can affirm that our protocol is safe regarding the specified security goals. It is impossible for an attacker to violate any of the specified security properties and disrupt the functioning of the protocol.

Following our formal validation, we focus, in the next section, on the energy cost savings achieved through our new exchange mode and our header compression scheme. The results are compared with the performances of the basic version of MIKEY-Ticket.

7.2 Performance analysis

As explained above, our contribution focuses on tailoring MIKEY-Ticket to the constrained environment of e-health applications. To this end, we propose a new header compression scheme along with a new exchange mode to reduce both the size of the exchanged messages and their number. In this subsection, we provide a performance analysis of our enhancements and compare energy consumption with the basic MIKEY-Ticket. First, we describe the energy model upon which our estimations are based. Then, we evaluate the communication and computational costs regarding both versions of MIKEY-Ticket (i.e. basic version and tailored version). The analysis is concluded with a discussion of the total energy cost highlighting the obtained energy savings.

7.2.1 Energy model and assumptions

Authors, in [MGSP08], have presented an energy evaluation of Wireless Sensor Nodes (WSN) regarding the communication cost. This latter is composed of the costs of transmission, reception and listening. Besides, the energy consumption of AES encryption algorithm and SHA-1 hash algorithm on WSN nodes have been also assessed in [KS06]. Both implementations were processed on tiny nodes with few MHz of computational power, several kilobytes of RAM and several tens of kilobytes of ROM.

In our evaluation, we consider the total energy cost as the sum of the communication cost and the computational cost. This latter is composed of encryption primitives based on AES and authentication primitives based on SHA-1 as specified in RFC 3830 [SO12c]. Based on the energy measurements presented in [MGSP08] and [KS06], we estimate the energy consumption of tiny nodes regarding both communication and computational aspects. The deduced values, summarized in Table 3, are used as an energy model of the different operations on constrained nodes.

```
user@instant-contiki:~/NewMikeyMode$ avispa  Mikey.hlpsl --ofmc
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
```

Figure 7: Avispa output (OFMC)

```
user@instant-contiki:~/NewMikeyMode$ avispa  Mikey.hlpsl --cl-atse

SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
```

Figure 8: Avispa output ($CL - AtSe$)

```
user@instant-contiki:~/NewMikeyMode$ avispa  Mikey.hlpsl  --satmc
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  BOUNDED_SEARCH_DEPTH
  BOUNDED_MESSAGE_DEPTH
```

Figure 9: Avispa output (SATMC)

```
user@instant-contiki:~/NewMikeyMode$ avispa  Mikey.hlpsl  --ta4sp
SUMMARY
   INCONCLUSIVE
```

Figure 10: Avispa output (TA4SP)

Transmission, reception, listening and cryptographic operations costs are considered for the evaluation of the total energy cost.

A set of assumptions is defined before diving into the details of our evaluation:

- Our evaluation only covers energy consumption of the constrained nodes as remote entities are not affected by resources scarcity. Hence, the efforts of reducing energy consumption are focused on the constrained part of the network model.
- In the estimation of message sizes, we only take into consideration the header part on which our compression scheme is applied. The other parts of the exchanged messages are constant regarding the two versions of MIKEY-Ticket.
- MIKEY specification has left the CS ID map info variable in length. In order to carry out our evaluation, we assume a 2 bytes long field.
- In order to evaluate the gains in energy savings of our compression scheme, we propose several levels of compression rates. These rates simulate different applications, each one defines a subset of fields to be compressed using our proposed 6LoWPAN-NHC-HDR. Table 4 presents the different compression rates along with the corresponding compressed fields.

7.2.2 Communication cost

- *Sending cost:* the sending cost is estimated by computing the overall size of the messages sent from the constrained node for both MIKEY-Ticket's versions. The cost is then computed for different levels of compression rate using the proposed energy model. Table 5 summarizes the results.

- *Receiving cost:* the receiving cost is estimated by computing the overall size of the messages sent to the constrained node for both MIKEY-Ticket's versions. The cost is then computed for different levels of compression rate using the proposed energy model. Table 6 summarizes the results.

- *Listening cost:* We consider the constrained node listening for a period of time equal to the sum of packets propagation delay (Δ), packets computation time (Comp), transmission latency (T) and reception latency (R). We assume the *KMS* being at one hop from the constrained node and 150 ms propagation delay needed for routing packets from the *KMS* to the remote entity. Moreover, we assume both *KMS* and *R* being 100 times more powerful than the tiny node *I* for the estimation of computational time. Furthermore, we consider, for the estimation of communication latency, an effective data rate of 75 *kbps* for a tiny node (e.g. TelosB) [MGSP08]. As an example, in the basic MIKEY-Ticket exchange mode, between the sending of *REQUEST_INIT* message and the reception of *REQUEST_RESP* message, the constrained node (CN) remains in the listening mode during the following period of time:

$$T_{listening} = R(KMS) + Comp(KMS) + T(KMS) + \Delta(KMS->CN).$$

Where:

- *R(KMS): Reception latency of KMS*
- *Comp(KMS): Computational time of KMS*
- *T(KMS): Transmission latency of KMS*
- $\Delta(KMS->CN)$*: Packets propagation delay from KMS to CN*

The cost is computed for different levels of compression rate, Table 7 summarizes the results. We notice a slight difference between the energy consumption of the two versions of MIKEY-Ticket. This is due to the fact that the listening time at the constrained node (i.e. I) is based on the time spent by the unconstrained nodes (i.e. KMS and R) to compute and communicate MIKEY-Ticket messages. In fact, their unconstrained resources make our tailoring's impact less visible.

7.2.3 Computational cost

- *Encryption cost:* the encryption cost is estimated by computing the overall size of the encrypted messages exchanged with the constrained node for both MIKEY-Ticket's versions. The cost is then computed for different levels of compression rate using the proposed energy model. Table 8 summarizes the results.

- *Authentication cost:* the authentication cost is estimated by computing the overall size of the messages exchanged with the constrained node on which a MAC is appended. The estimation is done regarding both MIKEY-Ticket's versions. The cost is then computed for different levels of compression rate using the proposed energy model. Table 8 summarizes the results.

Table 3 Estimated energy costs on constrained nodes

Operation	Cost
Transmit 1 bit	0.72 µJ
Receive 1 bit	0.81 µJ
Listen for 1 ms	0.29 µJ
AES-128 128-bits encryption	28.11 µJ
SHA-1 128-bits MAC computation	23.9 µJ

Table 4 Different compression rates

Compression rate (%)	Compressed fields	Gained space (bits)
0	None of the fields are compressed	0
16.4	V, DT	13
32.9	V, DT, PRF, MT	26
51.9	V, DT, PRF,♯ CS ,MI	41
72.1	V, DT, PRF, MT, CSB	57
83.5	V, DT,♯ CS, MI, CSB	66
100	All the fields are compressed	72

Table 5 Sending cost

	Compression rate (%)	Size (Bits)	Number of messages	Energy Cost (µJ)
Basic MIKEY-Ticket	0	96	02	138.24
Tailored MIKEY-Ticket	16.4	83	01	59.76
Tailored MIKEY-Ticket	32.9	70	01	50.4
Tailored MIKEY-Ticket	51.9	55	01	39.6
Tailored MIKEY-Ticket	72.1	39	01	28.08
Tailored MIKEY-Ticket	83.5	30	01	21.6
Tailored MIKEY-Ticket	100	24	01	17.28

Table 6 Receiving cost

	Compression rate (%)	Size (Bits)	Number of messages	Energy Cost (µJ)
Basic MIKEY-Ticket	0	96	02	155.52
Tailored MIKEY-Ticket	16.4	83	02	134.46
Tailored MIKEY-Ticket	32.9	70	02	113.4
Tailored MIKEY-Ticket	51.9	55	02	89.1
Tailored MIKEY-Ticket	72.1	39	02	63.18
Tailored MIKEY-Ticket	83.5	30	02	48.6
Tailored MIKEY-Ticket	100	24	02	38.88

Table 7 Listening cost

	Compression rate (%)	**Listening Time** (mS)	**Energy Cost** (μJ)
Basic MIKEY-Ticket	0	155.23	45.01
Tailored MIKEY-Ticket	16.4	153.32	44.5
Tailored MIKEY-Ticket	32.9	152.72	44.3
Tailored MIKEY-Ticket	51.9	152.1	44.1
Tailored MIKEY-Ticket	72.1	151.5	43.9
Tailored MIKEY-Ticket	83.5	151.2	43.8
Tailored MIKEY-Ticket	100	150.9	43.7

Table 8 Encryption cost

	Compression rate (%)	**Size (Bits)**	**Number of messages**	**Energy Cost** (μJ)
Basic MIKEY-Ticket	0	96	04	84.33
Tailored MIKEY-Ticket	16.4	83	03	54.68
Tailored MIKEY-Ticket	32.9	70	03	46.11
Tailored MIKEY-Ticket	51.9	55	03	36.23
Tailored MIKEY-Ticket	72.1	39	03	25.69
Tailored MIKEY-Ticket	83.5	30	03	19.76
Tailored MIKEY-Ticket	100	24	03	15.81

Table 9 Authentication cost

	Compression rate (%)	**Size (Bits)**	**Number of messages**	**Energy Cost** (μJ)
Basic MIKEY-Ticket	0	96	04	71.7
Tailored MIKEY-Ticket	16.4	83	03	46.49
Tailored MIKEY-Ticket	32.9	70	03	39.21
Tailored MIKEY-Ticket	51.9	55	03	30.80
Tailored MIKEY-Ticket	72.1	39	03	21.84
Tailored MIKEY-Ticket	83.5	30	03	16.80
Tailored MIKEY-Ticket	100	24	03	13.44

Table 10 Total energy cost

	Compression rate (%)	Communication cost	Computational cost	Total energy cost
Basic MIKEY-Ticket	0	338.77	156.03	494.8
Tailored MIKEY-Ticket	16.4	238.72	101.17	339.89
Tailored MIKEY-Ticket	32.9	208.1	85.32	293.42
Tailored MIKEY-Ticket	51.9	172.8	67.03	239.83
Tailored MIKEY-Ticket	72.1	135.16	47.53	182.69
Tailored MIKEY-Ticket	83.5	114	36.56	150.56
Tailored MIKEY-Ticket	100	99.86	29.26	129.11

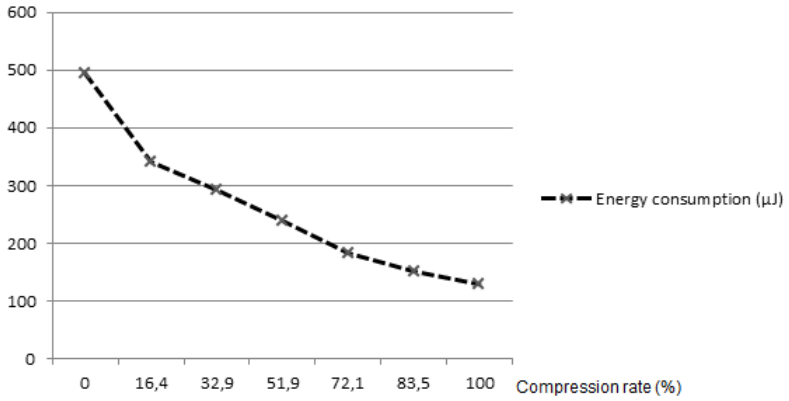

Figure 11: Total energy consumption on a constrained node for basic and tailored MIKEY-Ticket regarding different compression rates.

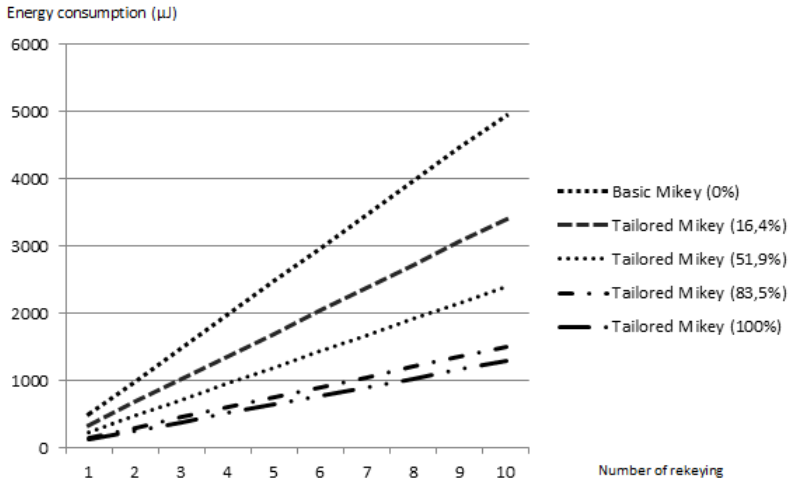

Figure 12: Energy consumption evolution through several rekeying operations for both MIKEY-Ticket versions regarding different compression rates

7.2.4 Discussion

Upon energy cost evaluation regarding both communication and computational aspects, we have estimated the overall energy cost considering both versions of MIKEY-Ticket. The results are synthesized in Table 10. As shown in Figure 11, we have already noticed a marked decrease at the first compression rate (i.e. 16,4%) due to the introduction of both new exchange mode and compression scheme which reduces the size and the number of the exchanged messages. Energy consumption keeps decreasing with the augmentation of compression rate. In fact, nearly 74% less energy is required to perform a full key exchange in the best case of our compression scheme.

The obtained results were expected as the reduction of both size and number of messages leads to a decrease in the energy spent either in the processing or in the communication of data. Nevertheless, an additional processing overhead is expected due to the compression/decompression operations of 6LoWPAN packets. As we consider the KMS being unconstrained, we can safely assume that the generated overhead will be supported by the KMS acting as a 6LoWPAN Border Router (6BR).

Additionally, we have compared the energy cost of several rekeying operations regarding different compression rates (see Figure 12). Frequent updates are likely to be performed in order to avoid long term attacks. The results show a considerable gain in the energy consumption that increases with the increase of rekeying operations. It is worth noticing that the gain is more important with the increase of rekeying operations which is critical for tiny nodes with highly constrained resources (e.g. increasing battery lifetime).

The analysis study allowed us to validate our proposition from two perspectives. First of all, we have provided a theoretical analysis regarding the different security properties required in our network scenario. The properties analysis has been validated using Avispa tool. Furthermore, we have proceeded with a quantitative analysis to highlight energy savings resulting from our tailoring of MIKEY-Ticket. Simulation showed the viability of the proposed solutions on e-health environments that are based on highly constrained sensor nodes. In a nutshell, our proposed solutions make MIKEY-Ticket more lightweight while its security properties are preserved.

8 Conclusion and future work

In this paper, we have introduced a tailoring mechanism for Mickey-Ticket to adapt it to low-power and constrained environment of e-health devices and applications. To this end, we have proposed a new header compression scheme to reduce the size of messages from 12 Bytes to 3 Bytes in the best compression case. In addition, we have introduced a new exchange communication mode to reduce the number of exchanged messages from six to four. We have evaluated our new solutions with respect to security and energy saving aspects. The results demonstrate that our approach keeps MIKEY-Ticket safe while considerable amount of energy is saved at the constrained node side. Hence, we can claim that our adjustments of MIKEY-Ticket protocol are well-adapted to IoT constrained environments such as e-health applications. In the future, we are going to investigate the applicability of our tailored MIKEY-Ticket for group communication scenarios, and the eventual impact of mobility on the architectural entities.

References

[AIM10] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, May 2010.

[ALK12] Moshaddique Al Ameen, Jingwei Liu, and Kyungsup Kwak. Security and privacy issues in wireless sensor networks for healthcare applications. *J. Med syst*, 36:93–101, 2012.

[ALNN04] J. Arkko, F. Lindholm, M. Naslund, and K. Norrman. Mikey: Multimedia internet keying. *RFC 3830, IETF*, 2004.

[BBL+12] R. Bonetto, N. Bui, V. Lakkundi, A. Olivereau, A. Serbanati, and M. Rossi. Secure communication for smart iot objects: Protocol stacks, use cases and practical examples. *In Proc. of IEEE WoWMoM, 2012*, 2012.

[BOO13] Aymen Boudguiga, Alexis Olivereau, and Nouha Oualha. Server assisted key establishment for wsn: A mikey-ticket approach. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, pages 94–101. IEEE, 2013.

[CCC+04] Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, J. Mantovani, and and L. Vigneron S. Modersheim. A high level protocol specification language for industrial security sensitive protocols. *Proc. SAPS 04. Austrian Computer Society, 2004*, 2004.

[CDS+11] C. Chun, H. Daojing, C. Sammy, B. Jiajun, G. Yi, and F. Rong.

Lightweight and provably secure user authentication with anonymity for the global mobility network. *International Journal of Communication Systems*, 24(3):347–362, 2011.

[CGP99] EM. Clarke, O. Grumberg, and DA. Peled. Model checking. *MIT Press: Cambridge*, 1999.

[CM09] A. Charu and T. Mathieu. Validating integrity for the ephemerizers protocol with cl-atse. pages 21–32, 2009.

[CVG03] S. Cherukuri, K.K. Venkatasubramanian, and S.K.S. Gupta. Biosec: a biometric based approach for securing communication in wireless networks of biosensors implanted in the human body. *Parallel Processing Workshops Proceedings, International Conference, 2003*, October 2003.

[DMOD+] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier. The internet of things for ambient assisted living. In *Information Technology: New Generations (ITNG), 2010*.

[Dou02] J.R. Douceur. The sybil attack. *Peer-to-peer Systems*, 2002.

[DY81] D. Dolev and C.C. Yao. On the security of public key protocols. *FOCS, IEEE, 1981*, 1981.

[ER11] N. Modadugu E. Rescorla. Datagram transport layer security version 1.2. *RFC 6347, IETF*, 2011.

[FHM+07] T. Freeman, R. Housley, A. Malpani, D. Cooper, and W. Polk. Server-based certificate validation protocol(scvp). *RFC 5055, IETF, 2007*, 2007.

[GG06] Y. Glouche and T. Genet. Span a security protocol animator for avispa user manual. *http ://www.irisa.fr/lande/genet/span/, 2006*, 2006.

[GMS10] J. Granjal, E. Monteiro, and J. Sa Silva. Enabling network-layer security on ipv6 wireless sensor networks. *Proc. of IEEE GLOBECOM, 2010*, 2010.

[HHHW13] Ren Hummen, Jens Hiller, Martin Henze, and Klaus Wehrle. Slimfit - a hip dex compression layer for the ip-based internet of things. *WiMob, IEEE, 2013*, 2013.

[HHW+13] R. Hummen, J. Hiller, H. Wirtz, M. Henze, H. Shafagh, and K. Wehrle. 6lowpan fragmentation attacks and metigation mechanisms. *Proc. 6th ACM Conf. Security Privacy Wireless Mobile Networks*, pages 55–66, Apr 2013.

[HNL08] Michael Healy, Thomas Newe, and Elfed Lewis. *Smart Sensors and Sensing Technology*, chapter Analysis of Hardware Encryption Versus Software Encryption on Wireless Sensor Network Motes. Springer Berlin Heidelberg, 2008.

[HRZ08] Y. Hanna, H. Rajan, and W. Zhang. Slede: A domain specific verification framework for sensor network security protocol implementations. *Proceeding of the ACM Conference on Wireless Network Security (WiSec08)*, pages 109–118, 2008.

[HT11] J. Hui and P. Thubert. Compression format for ipv6 datagrams over ieee 802.15.4-based networks. *RFC 6282, IETF, 2011*, 2011.

[HWZ+13] Rene Hummen, Hanno Wirtz, Jan Henrik Ziegeldorf, Jens Hiller, and Klaus Wehrle. Tailoring end-to-end ip security protocols to the internet of things. *in Proc. of IEEE ICNP, 2013*, 2013.

[HZS+13] Ren Hummen, Jan H. Ziegeldorf, Hossein Shafagh, Shahid Raza, and Klaus Wehrle. Towards viable certificate-based authentication for the internet of things. *HotWiSec '13 Proceedings of the 2nd ACM workshop on Hot topics on wireless network security and privacy, 2013*, 2013.

[IJSP10] R. Istepanian, A. Jara, A. Sungoor, and N. Philips. Internet of things for m-health applications (iomt). *AMA-IEEE medical technology conference on individualized healthcare,Washington, 2010*, 2010.

[JR13] S.S. Javadi and M.A. Razzaque. Security and privacy in wireless body area networks for health care applications. *Wireless Networks and Security*, pages 165–187, 2013.

[KHNE10] C. Kaufman, P. Hoffman, Y. Nir, and P. Eronen. Internet key exchange protocol version 2 (ikev2). *RFC 5996, IETF*, 2010.

[KKT14] S.L. Keoh, S.S. Kumar, and H. Tschofenig. Securing the internet of things: A standardization perspective. *IEEE INTERNET OF THINGS JOURNAL*, 2014.

[KS06] J.P. Kaps and B. Sunar. Energy comparison of aes and sha-1 for ubiquitous computing. *Emerging Directions in Embedded and Ubiquitous Computing, Lecture Notes in Computer Science, 2006*, 2006.

[KSW04] C. Karlof, N. Sastry, and D. Wagner. Tinysec: A link layer security architecture for wireless sensor networks. *Second ACM Conference on Embedded Networked Sensor Systems, 2004*, november 2004.

[LL10] Ming Li and Wenjing Lou. data security and privacy in wireless body area networks. *Wireless Technologies for E-healthcare, 2010*, February 2010.

[LLL+12] A. Le, J. Loo, A. Lasebae, M. Aiash, and Y. Luo. 6lowpan: a study on qos security threats and countermeasures using intrusion detection system approach. *International Journal of Communication Systems*, 25(9), 2012.

[LOCL10] S. Lim, T.H. Oh, Y.B Choi, and T. Lakshman. Security issues on wireless body area network for remote healthcare monitoring. *Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), IEEE International Conference*, pages 327 – 332, February 2010.

[MC11] M. Marino and U. Caterina. Formal analysis of facebook connect single sign-on authentication protocol. 11:22–28, 2011.

[MD03] S. Moedersheim and P.H. Drielsma. Avispa project deliverable d6.2: Specification of the problems in the high-level specification language. http://www.avispa-project.org, 2003.

[MGSP08] G. De Meulenaer, F. Gosset, F.X Standaert, and O. Pereira. On the energy cost of communication and cryptography in wireless sensor networks. *IEEE International Conference on Wireless and Mobile Computing, Networking and Communication, 2008*, 2008.

[MKHC07] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler. Transmission of ipv6 packets over ieee 802.15.4 networks. *RFC 4944, IETF, 2007*, 2007.

[MRL06] S.S.M. Meingast, T. Roosta, and Elfed Lewis. Security and privacy issues with health-care information technology. *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2006*, 2006.

[MT11] J. Mattsson and T. Tian. Mikey-ticket: Ticket-based modes of key distribution in multimedia internet keying (mikey). *RFC 6043, IETF, 2011*, 2011.

[NST06] H. S. Ng, M.L. Sim, and C.M. Tan. Security issues of wireless sensor networks in healthcare applications. *BT Technology Journal*, 24(2):138–144, 2006.

[PW10] M. Patel and J. Wang. Applications, challenges, and prospective in emerging body area networking technologies. *Wireless Commun, 2010*, 2010.

[PZB06] C.C.Y. Poon, Yuan-Ting Zhang, and Shu-Di Bao. A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health. *Communications Magazine, IEEE, 2006*, 4, April 2006.

[RALS11] Rodrigo Roman, Cristina Alcaraz, Javier Lopez, and Nicolas Sklavos. key management systems for sensor networks in the context of internet of things. *Computers and Electric Engineering, 2011*, 2011.

[RDC+11] S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt, and U. Roedig. Securing communication in 6lowpan with compressed ipsec. *in Proc. of IEEE DCOSS, 2011*, 2011.

[RDS13] S. Raza, S. Duquennoy, and G. Selander. Compression of ipsec ah and esp headers for constrained environments. *draft-raza-6lowpanipsec-00 (WiP), IETF, 2013*, 2013.

[refa] Avispa a tool for automated validation of internet security protocols. *http://www.avispa-project.org*.

[refb] Murphi model checker. *http://www.cs.utah.edu*.

[refc] Prism - a probabilistic model checker. *http://www.prismmodelchecker.org*.

[RMMLBLS06] Antonio Ruiz-Martínez, C. Inmaculada Marín-López, Laura Baño-López, and AF Skarmeta. A new fair non-repudiation protocol for secure negotiation and contract signing. page 16, 2006.

[RTV12] S. Raza, D. Trabalza, and T. Voigt. 6lowpan compressed dtls for coap. *in Proc. of IEEE DCOSS, 2012*, 2012.

[RVJ12] Shahid Raza, Thiemo Voigt, and Vilhelm Jutvik. Lightweight ikev2: A key management solution for both compressed ipsec and ieee 802.15.4 security. *IETF/IAB workshop on Smart Object Security, 2012*, 2012.

[RWV13] S. Raza, L. Wallgren, and T. Voigt. Svelte: Real-time intrusion detection in the internet of things. *Ad hoc networks*, 11(8), 2013.

[SO12a] Y. B. Saied and A. Olivereau. D-hip: A distributed key exchange scheme for hip-based internet of things. *in Proc. of IEEE WoWMoM, 2012*, 2012.

[SO12b] Y. B. Saied and A. Olivereau. Hip tiny exchange (tex): A distributed key exchange scheme for hip-based internet of things. *in Proc. of ComNet, 2012*, 2012.

[SO12c] Y. B. Saied and A. Olivreau. (k, n) threshold distributed key exchangefor hip based internet of things. *in Proc. of ACM MobiWac, 2012*, 2012.

[TCC+09] L. Tobarra, D. Cazorla, F. Cuartero, G. Diaz, and E. Cambronero. Model checking wireless sensor network security protocols: Tinysec + leap + tinypk. *Telecommunication Systems*, 40(3-4):91–99, 2009.

[TD11] N. Tsiftes and A. Dunkels. A database in every sensor. *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems, 2011*, 2011.

[WGE+05] A.S. Wander, N. Gura, H. Eberle, V. Gupta, and S.C. Shantz. Energy analysis of public-key cryptography for wireless sensor networks. *IEEE Pervasive Computing and Communications*, 2005.