A Lightweight Authentication Mechanism for M2M Communications in Industrial IoT Environment

Alireza Esfahani, Georgios Mantas, Rainer Matischek, Firooz B. Saghezchi, Jonathan Rodriguez, Ani Bicaku, Silia Maksuti, Markus Tauber, Christoph Schmittner, and Joaquim Bastos

***Abstract*—In the emerging Industrial IoT era, Machine-to-Machine (M2M) communication technology is considered as a key underlying technology for building Industrial IoT environments where devices (e.g., sensors, actuators, gateways) are enabled to exchange information with each other in an autonomous way without human intervention. However, most of the existing M2M protocols that can be also used in the Industrial IoT domain provide security mechanisms based on asymmetric cryptography resulting in high computational cost. As a consequence, the resource-constrained IoT devices are not able to support them appropriately and thus, many security issues arise for the Industrial IoT environment. Therefore, lightweight security mechanisms are required for M2M communications in Industrial IoT in order to reach its full potential. As a step towards this direction, in this paper, we propose a lightweight authentication mechanism, based only on hash and XOR operations, for M2M communications in Industrial IoT environment. The proposed mechanism is characterized by low computational cost, communication and storage overhead, while achieving mutual authentication, session key agreement, device's identity confidentiality, and resistance against the following attacks: replay attack, man-in-the-middle attack, impersonation attack, and modification attack.**

***Index Terms*—Industrial IoT, M2M communications, Lightweight Authentication, Security, Sensors.**

I. Introduction

THE Industrial Internet of Things (IIoT) technology is a key enabler for the next industrial revolution, known as Industry 4.0 [1], [2]. The invention of the steam engine by James Watt in the 18th century caused the first generation of industrial production; the invention of electric power brought about the second industrial revolution in 1870 with the widespread use of electric machines in production lines [3]. The industrial automation and widespread adoption of computers and programmable logic controllers (PLC) in 1970s staged the third industrial revolution. Nowadays, IIoT and Machine-to-Machine (M2M) communications are about to bring the fourth industrial revolution known as Industry 4.0 [4]–[6], where man, machine, and product will be interconnected throughout the whole supply-chain from the production floor to the managerial level. This will boost the productivity and allow customised and flexible production while benefiting from the economies of scale [7], [8].

However, the transition from the third industrial revolution to Industry 4.0 raises a wide spectrum of new security issues [3], [5], [9]–[11]. Traditional industrial communication systems have been designed for reliable operation in a noisy factory environment, employing mainly hard-wired propriety-based communication technologies to connect sensors, actuators, and controllers as well as other industrial components such as Supervisory Control and Data Acquisition System (SCADA) and Manufacturing Execution System (MES). Nevertheless, with the emergence of IIoT, future factories will increasingly rely on diverse communication technologies including wireless standards to ensure connectivity, interoperability, and remote operation and control of production processes through the Internet. This provides an unprecedented attack surface for the attackers. Unlike computer networks, where an attack normally threatens the information integrity, confidentiality, or availability, attacks against a smart factory can cause physical damage, threaten the human life, render the quality or final products by compromising the production processes, or lead to increased use of resources [9]. Last but not least, unlike the short lifetime of consumer electronics, the lifespan of machines operating on a production floor normally lasts for several decades, and it is not always economically viable to completely replace the legacy equipment with the latest technology. Hence, it is essential to come up with novel solutions that ensure security not only for the leading-edge manufacturing technology but also for the legacy systems.

Since authentication is the cornerstone of providing effective security, a number of authentication schemes have been proposed to ensure security in IoT or M2M applications [12]–[16]; however, they cannot be readily applied for IIoT because manufacturing machines are naturally limited with computation power and/or communication bandwidth. In this paper, we propose a lightweight authentication scheme to authenticate these resource constrained machineries in order to ensure secure integration of IIoT solutions in the future production systems. To this end, we consider an IIoT scenario where a machine (i.e., a smart sensor), equipped with a Secure Element (SE), is authenticated by a network element (i.e., a router) equipped with a Trusted Platform Module (TPM).

The rest of this paper is organized as follows. Section II reviews the related work. Section III describes the proposed authentication mechanism. Section IV presents the security

Alireza Esfahani, Georgios Mantas, and Joaquim Bastos are with the Instituto de Telecomunicações - Pólo de Aveiro, Aveiro, Portugal (e-mail: alireza@av.it.pt; gimantas@av.it.pt; jbastos@av.it.pt).

Rainer Matischek is with the Infineon Technologies Austria AG Graz, Austria (e-mail: rainer.matischek@infineon.com).

Firooz B. Saghezchi and Jonathan Rodriguez are with the University of Aveiro, Aveiro, Portugal (e-mail: firooz@ua.pt; jonathan@ua.pt).

Ani Bicaku, Silia Maksuti, and Markus Tauber are with the University of Applied Sciences, Burgenland, Eisenstadt, Austria (e-mail: ani.bicaku@fh-burgenland.at; silia.maksuti@fh-burgenland.at; markus.tauber@fh-burgenland.at).

Christoph Schmittner is with the Austrian Institute of Technology, Wien, Austria (e-mail: Christoph.Schmittner@ait.ac.at).

analysis for the proposed mechanism, and Section V discusses its performance evaluation in terms of communication overhead, computational cost, and storage cost. Finally, Section VI concludes.

II. Related Work

A. M2M Communication protocols for IIoT

The main focus of communication protocols is to guarantee the delivery, routing and storage of the information without the necessity of implementing different mechanisms in different devices or applications. Based on relevant scientific works, we evaluate the most popular protocols for use in IIoT and M2M communication. As a result, according to [33] and [34] the most discussed and promoted standards for the communication between devices and cloud services are:

- **6LoWPAN: IPV6 over Low power Wireless Personal Area Network**
  An Internet Protocol version 6 (IPv6) over low-power wireless personal area networks (6LoWPAN) standard has been developing, by the Internet Engineering Task Force (IETF), in order to promote the development of the IoT and exploit the M2M applications. 6LoWPAN enables IP-based M2M devices to connect to the Internet. More precisely, one potential application is to monitor the manufacture process in industry [26], [27], where a number of sensors, actuators, and controllers are connected together to achieve passive monitoring and active control and automation. However, 6LoWPAN has various security challenges and many threats and trust crises are existing along with its development. Sensor nodes are usually distributed in an unprotected environment and messages can be easily eavesdropped in the transmission. A lot of research works have been proposed to overcome the vulnerabilities of the 6LoWPAN systems. A secure authentication and key establishment scheme (SAKES) has been proposed in [17]. SAKES makes use of two different cryptographic schemes. Firstly, the authors propose the use of a symmetric key scheme to encrypt messages in the authentication phase and then, an asymmetric key scheme based on the elliptic curve cryptography (ECC) is used, in the key establishment phase, to build a session key between the 6LoWPAN devices and the server. More precisely, the session key in the SAKES scheme is calculated based on the Diffie–Hellman (DH) key exchange method. EAKES6Lo scheme proposed by Qiu et.al is a mutual authentication and key establishment scheme for M2M communication in 6LoWPAN Networks [18]. It consists of three phases: pre-deployment phase, AKE phase, and handover phase. Their security analysis shows that the proposed scheme can be resistant against replay attacks, Man-in-the-Middle attacks, impersonation attacks, Sybil attacks, and compromised attacks. Moreover, it can benefit from low computational overhead and transmission overhead.

- **MQTT: Message Queue Telemetry Transport**
  Message Queue Telemetry Transport, is an OASIS standardized protocol [19]. It is designed to be lightweight, flexible and simple to implement. MQTT uses different routing mechanisms, such as: one-to-one; one-to-many or many-to-many, making possible the connection for IoT and M2M to connected devices/applications. MQTT is a publish/subscribe messaging transport protocol on the top of TCP/IP protocol consisting of three components (subscriber, publisher, and broker). In terms of Quality of Service (QoS), it supports three levels: (i) QoS-0:The message will be delivered once, with no confirmation, (ii) QoS-1: The message will be delivered at least once, with confirmation required, (iii)QoS-2: The message will be delivered exactly once by using a handshake. However, although MQTT has been deployed for IoT, it has limited security features addressing IoT security issues. In particular, MQTT uses user-name/password authentication and SSL/TLS for secure data communication [20]. Hence, further efforts are required so that MQTT can address effectively and efficiently security issues for IoT. In [11], the authors have envisaged the use of SSL/TLS with certificates and session key management to enhance security for MQTT. In particular, they have proposed a Secure MQTT (SMQTT). Their proposed solution is based on lightweight Attribute Based Encryption (ABE) [22], [23] over elliptic curves [35]. They have used ABE because of its inherent design supporting broadcast encryption that is suitable for IoT applications. ABE includes two types: (i) Cipher-text Policy based ABE (CP-ABE), and (ii) Key Policy based ABE (KPABE). Authors' analysis and performance evaluation show that SMQTT is efficient, robust, and scalable.

- **AMQP: Advanced Message Queuing Protocol**
  Advanced Message Queuing Protocol, is a binary application layer protocol standardized from the ISO/IEC 19464. It is a message centric protocol on top of TCP/IP, which provides publish-subscribe and point-to-point communication. AMQP supports message-oriented communication via message-delivery guarantees including: (i) at-most-once, when each message is delivered once or never, (ii) at-least- one, when each message is delivered and (iii) exactly-one, when the message will certainly delivered only once [24]. AMQP provides different features, including routing and storing messages within the broker using message queues. In terms of security, it supports SASL authentication and TLS for secure data communication [25].

- **CoAP: Constrained Application Protocol**
  Constrained Application Protocol, is a web transfer protocol which supports unicast and multicast requests for use in constrained devices and networks. It is based on a request-response architecture between endpoints. CoAP clients after sending the requests using an URI, can receive as a response GET, PUT, POST and DELETE resources from the server [26]. The messages are

TABLE I
OVERVIEW OF POSSIBLE M2M PROTOCOLS FOR IIOT

Protocol	Feature	Layer	TCP/UDP	Security	Related work
6LOWPAN	To map services required by the IPv6 over Low power WPANs to maintain an IPv6 network	Network	TCP	SSL	[17], [18]
MQTT	To utilize the publish/subscribe pattern to provide transition flexibility and simplicity of implementation	Application	TCP	SSL	[19]–[23]
AMQP	To provide publish-subscribe and point-to-point communication	Application	TCP	SSL	[24], [25]
CoAP	To connect resource-constrained devices in a secure and reliable way	Application	UDP	DTLS	[26]–[29]
XMPP	To transfer instant messaging (IM) standard that is used for multi-party chatting, voice and video calling and telepresence	Application	TCP	SSL	[30], [31]
DSS	To enable scalable, real-time, dependable, high-performance and interoperable data exchanges using a publish–subscribe pattern	Application	TCP/UDP	SSL	[32], [33]

exchanged over UDP between endpoints and also it supports the use of unicast and multicast requests. In terms of QoS, CoAP supports two levels: (i) 'confirmable' when no packet is lost and the receiver respond with an ACK; and (ii) 'non-confirmable' when the message do not require an ACK. CoAP provides security via the Datagram Transport Layer Security (DTLS) which is a secure protocol for network traffic to support handling packet loss, reordering of messages and message size [27]. However, DTLS requires numerous message exchanges to establish a secure session and thus it is characterized by high communication cost. Therefore, CoAP suffers from this challenge of DTLS. In order to overcome this issue, a lightweight secure CoAP for the Internet of Things (Lithe) scheme has been proposed by Raza et.al [28]. Lithe scheme shows that DTLS can be compressed and its overhead is significantly reduced. The proposed scheme is implementated in Contiki [29] and their evaluation shows significant gains in terms of packet size, energy consumption, processing time, and network-wide response times when compressed DTLS is enabled.

- **XMPP: Extensible Messaging and Presence Protocol**
  Extensible Messaging and Presence Protocol, is a TCP communication protocol based on Extensible Markup Language (XML) used for real-time messaging, online presence and request-response services [30]. Clients communicate via a distributed network and do not rely on a central broker. XMPP supports publish/subscribe model and provides security such as authentication via SASL and secure communication via TLS but does not provide any level of QoS [31].

- **DDS: Data Distribution Service**
  Data Distribution Service (DDS) is one of the publish-subscribe protocol for real-time M2M communications which has been developed by Object Management Group (OMG) [32]. In contrast to other publish-subscribe application protocols like MQTT or AMQP, DDS relies on a broker-less architecture and uses multicasting. These facilities provide high reliability to its applications. Its broker-less publish-subscribe architecture suits well to the real-time constraints for IoT and M2M communications. DDS defines a comprehensive set of QoS policies. These provide control over dynamic discovery, content-aware routing and filtering, fault tolerance and deterministic real-time behavior [33].

B. TPM (Trusted Platform Module)

Trusted Platform Module (TPM) is an international standard for secure cryptographic processors written by Trusted Computing Group (TCG), documented as ISO/IEC 11889 standard. OPTIGA^{TM} TPM is a portfolio of security chips (cryptocontroller), provided by Infineon Technologies AG, to protect integrity and authenticity of embedded devices and systems, carry out remote attestation, and perform cryptographic functions. It supports both TPM 1.2 and TPM 2.0 standards and provides a secure communication channel between smart factories to protect data, processes, and intellectual property against potential sabotage or theft. Its key features include secured key, certificate, and password storage as well as dedicated key management and support for a variety of encryption algorithms [36], [37].

III. PROPOSED AUTHENTICATION MECHANISM

In this Section, we propose a lightweight authentication mechanism for machine-to-machine communication between a resource-constrained industrial device (e.g., smart sensor) including a Secure Element (SE) and a router including a TPM. The proposed mechanism is inspired by [38] and includes two procedures: a) the registration procedure where the sensor is registered to the Authentication Server (AS) and b) the authentication procedure where mutual authentication between the sensor and the router is achieved. The used notations in the proposed mechanism are listed in Table II.

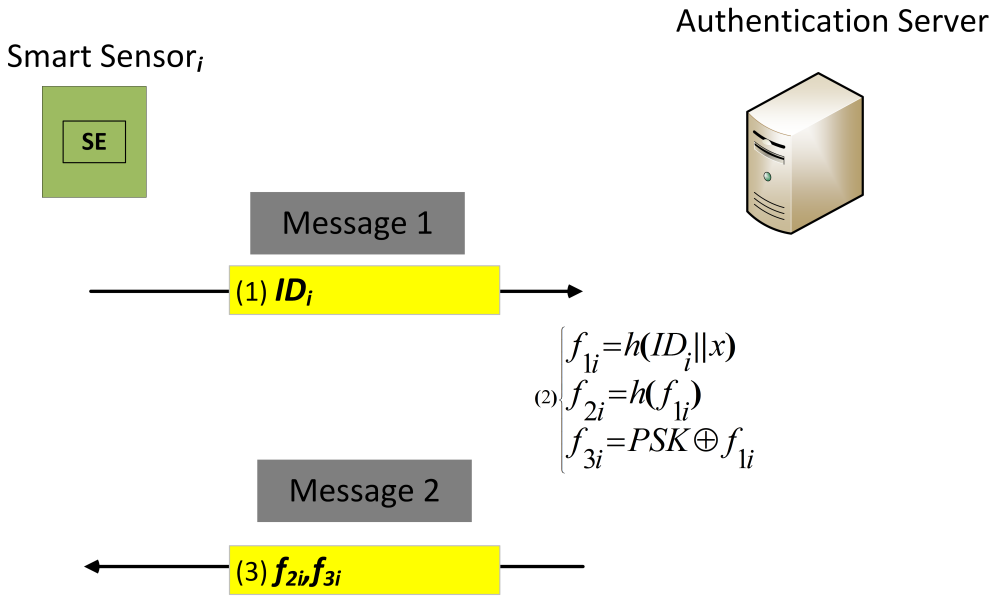

Fig. 1. Registration procedure.

TABLE II
NOTATIONS

Symbol	Description
x	A secret key protected by the AS
PSK	A secure pre-shared key between the AS and the router
ID_i	The identity of smart sensor i
AID_i	The alias of entity i
f_i	Function generation
SK_i	The shared key between smart sensor i and a router
R_i	A random number generated by a Pseudorandom Number Generator (PRNG)
$h(.)$	A one-way hash function
$\|$	A concatenation operator
$\oplus$	XOR operation

A. Registration

Each smart sensor needs to perform the registration procedure with the AS through a secure channel. The AS generates the secure pre-shared key set $PSK_i, i = 1, ..., n$ and sends each PSK_i to one of the routers. The registration procedure consists of the following steps, as it is shown in Figure 1.

1) Smart sensor $\rightarrow$ AS: The smart sensor transmits its unique identity number ID_i to the AS over a secure channel.
2) Upon receiving the smart sensor's ID, the AS calculates the following three secret authentication parameters for the sensor: $f_{1i} = h(ID_i \parallel x)$, $f_{2i} = h(f_{1i})$, and $f_{3i} = PSK \oplus f_{1i}$. The objective of f_{1i}, f_{2i} is to build the relation between sensor's ID and AS.
3) AS $\rightarrow$ Smart sensor: The AS sends the following parameters in the smart sensor via a secure channel: f_{2i}, f_{3i}. These parameters are stored in the SE of the sensor.

B. Authentication

After the registration phase, each sensor is able to authenticate to a router. It is worthwhile to mention that, during the authentication procedure, the sensor never uses its real identity to authenticate to a router. Hence, the smart sensor's ID cannot be eavesdropped by a malicious entity. The authentication procedure consists of the following steps, as it is shown in Figure 2.

1) The smart sensor generates a random number R_1 and stores it in the SE of the sensor. Then, it computes the parameter M_1 as follows: $M_1 = h(f_{2i}) \oplus R_1$. Afterwards, the sensor computes its alias as $AID_i = h(R_1) \oplus ID_i$ and computes the parameter M_2 as follows: $M_2 = h(R_1 \parallel M_1 \parallel AID_i)$.
2) Smart sensor $\rightarrow$ router: The smart sensor sends to the router an authentication request (i.e., Message 3) including $(M_1, M_2, f_{3i}, AID_i)$.
3) Upon receiving the authentication request, the router performs the following. Firstly, the router retrieves f_{1i} by using the pre-shared key PSK (i.e., $f_{1i} = f_{3i} \oplus PSK$). Then, the router obtains R_1 and ID_i via $R_1 = M_1 \oplus h(f_{2i})$ and $ID_i = AID_i \oplus h(R_1)$, respectively. Then, router checks whether $h(R_1 \parallel M_1 \parallel AID_i)$ is equal to M_2. The authentication request is rejected if $h(R_1 \parallel M_1 \parallel AID_i)$ and M_2 do not match. Next, the router generates a random number R_2 that is stored in the TPM of the router. Then, it computes AID_j, $M_1^{'}$, and $M_2^{'}$ as follows: $AID_j = R_2 \oplus h(ID_i)$, $M_1^{'} = f_{1i} \oplus h(ID_i)$, and $M_2^{'} = h(M_1^{'} \parallel AID_j \parallel R_2)$. Finally, the router calculates the session key SK_{ij} as $SK_{ij} = h(R_1 \parallel R_2)$.
4) Router $\rightarrow$ Smart sensor: The router sends back to the sensor the authentication response (Message 4) including $M_1^{'}$, $M_2^{'}$ and AID_j.

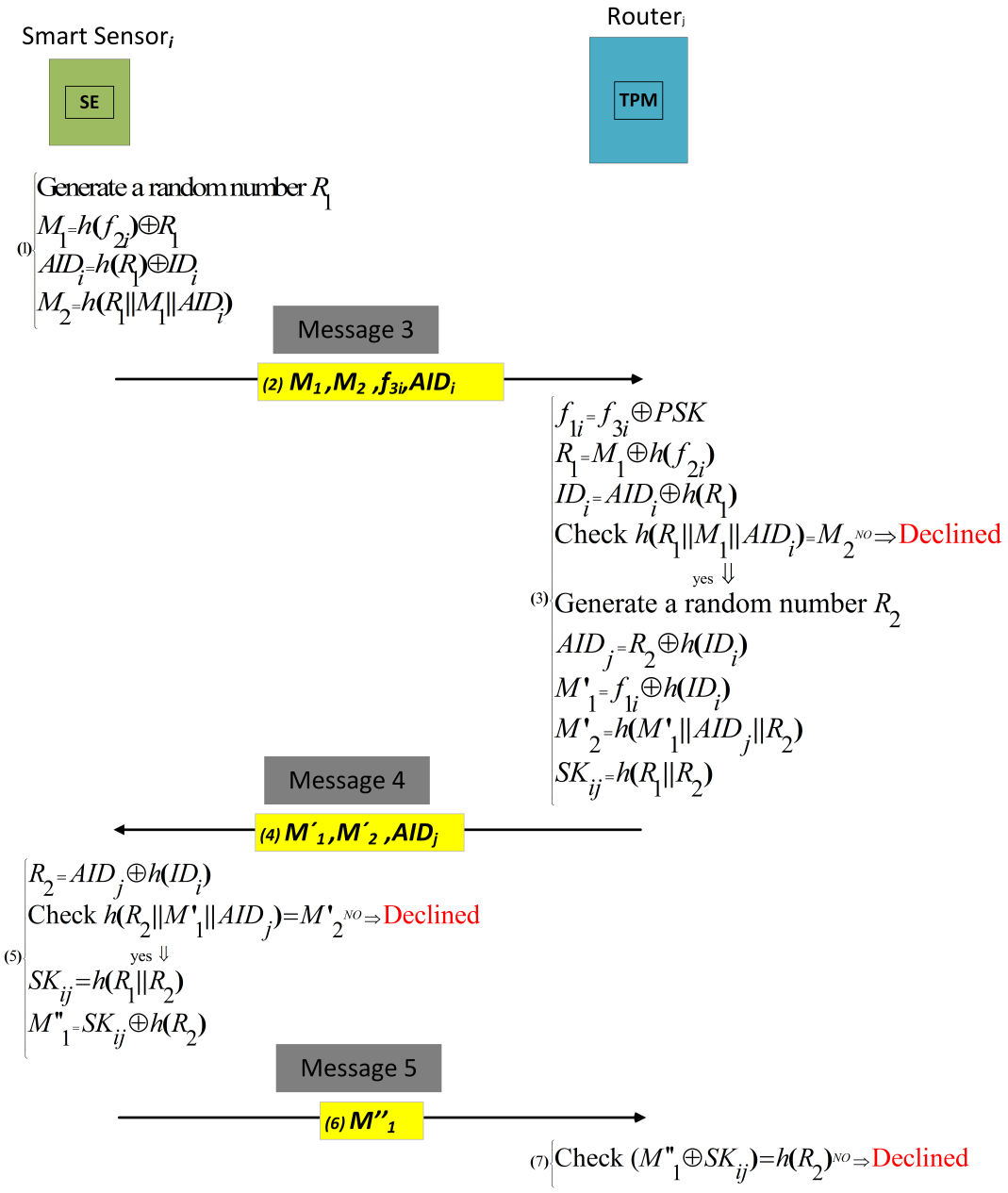

Fig. 2. Authentication procedure.

5) The smart sensor retrieves R_2 by computing $AID_j \oplus h(ID_i)$ and checks if $h(R_2 \parallel M_1^{'} \parallel AID_j)$ and $M_2^{'}$ are equal. If they are equal, the router calculates the session key SK_{ij} as $SK_{ij} = h(R_1 \parallel R_2)$. Finally, the smart sensor calculates $M^{''}{}_1$ as $SK_{ij} \oplus h(R_2)$.
6) Smart sensor $\rightarrow$ Router:The smart sensor sends Message 5 including $M^{''}{}_1$ to the router.
7) Upon receiving Message 5, the router uses its session key SK_{ij} calculated in step 3 in order to retrieve $h(R_2)$. Then, the router calculates $SK_{ij} \oplus M^{''}{}_1$ and compares it with $h(R_2)$. If they are equal, it means that the smart sensor holds the legitimate session key.

IV. Security Analysis

In this Section, we provide the security analysis of the proposed authentication mechanism. We have adopted the security analysis approach followed in [39]–[41] and thus, we have the following:

Proposition 1. *Smart sensor's identity confidentiality is provided by the proposed mechanism.*

Proof. In the proposed mechanism, the confidentiality of the smart sensor's identity is based on a hash value of a random number R_1 and a XOR function (i.e., $AID_i = h(R_1) \oplus ID_i$). Therefore, the adversary cannot derive the identity ID_i of the smart sensor without knowing the random number R_1. □

Proposition 2. *The proposed mechanism provides entity mutual authentication.*

Proof. In the authentication phase, mutual authentication between the smart sensor and the router can be achieved based on the received Message 3 and Message 4. Upon receipt of M_1, M_2, f_{3i}, and AID_i, the router checks whether M_2 is equal to $h(R_1 \parallel M_1 \parallel AID_i)$. The smart sensor is considered authenticated if the equality holds. The same process takes place in authenticating the router when the smart sensor receives $M_1^{'}$, $M_2^{'}$ and AID_j. The smart sensor computes $h(R_2 \parallel M_1^{'} \parallel AID_j)$ and checks whether this value is equal to $M_2^{'}$. If they are equal, the router is also considered as authenticated. Moreover, if the adversary aims to forge a valid smart sensor/router, he/she needs to generate valid messages. However, the adversary cannot generate the valid messages because he has no information about the random numbers (i.e., R_1 and R_2). □

Proposition 3. *The proposed mechanism is resistant to replay attack.*

Proof. As it is described in Subsection III-B, we assume that a legitimate smart sensor has sent Message 3 (i.e., M_1, M_2, f_{3i}, and AID_i) to the router. If an adversary tries to impersonate the legitimate smart sensor by replaying Message 3, the router will reject the authentication request because the alias AID_i of the smart sensor is calculated based on a hash value of a random number R_1 which is only known to the legitimate sensor. □

Proposition 4. *The proposed mechanism is resistant to man-in-the-middle attack.*

Proof. By obtaining the smart sensor's identity (i.e., ID_i), an adversary is not able to lunch a man-in-the-middle attack and computes a session key SK_{ij}, because he/she is not able to obtain the secret key x that is only known to the AS. x is securely stored in the authentication server and is never transmitted to any other entity. Meanwhile, the adversary cannot pretend that is a trustful router to authenticate other smart sensors since he/she does not have the pre-shared secret key PSK. □

Proposition 5. *The proposed mechanism is resistant to impersonation attack.*

Proof. We consider that a smart sensor (e.g., $smartsensor_k$) intends to impersonate another smart sensor (e.g., $smartsensor_i$). However, $smartsensor_k$ cannot obtain the session key SK_{ij} generated by the $smartsensor_i$ and the router because it has no information about the random number of $smartsensor_i$. □

Proposition 6. *The proposed mechanism is resistant to modification attack.*

Proof. The one-way hash function $h()$ guarantees that information cannot be modified without being detected. If an adversary transmits a modified message to the router, the router will detect it by checking the hash values. □

TABLE III
SETTING OF PARAMETERS

Parameters	Value (bits)
ID_i	128
AID_i	128
Random number	128
Hash value	128
f_{1i}	128
f_{2i}	128
f_{3i}	128

Moreover, the proposed authentication mechanism can achieve the following security objectives:

1) **No clock synchronization:**
   In the proposed mechanism, it is not required to synchronize the clock of the devices (e.g., smart sensor, router, and authentication server) because the messages which are exchanged between the devices are similar to the messages exchanged in the nonce-based authentication mechanism [38] which does not rely on timestamps.
2) **Fast error detection:**
   In the authentication procedure, the router will detect an error immediately if an adversary uses the wrong sensor ID. This means that a legitimate smart sensor will have been identified by the router before the calculation of session key.
3) **Independent session key:**
   If the session key SK_{ij} is compromised by an adversary, the smart sensor and the router can generate a new session key. This is because the generation of the session key SK_{ij} in the proposed authentication mechanism is based on a hash function and random numbers and thus, it is independent to the previous session key.

V. PERFORMANCE EVALUATION

In this Section, we evaluate the performance of the proposed authentication mechanism in terms of communication overhead, computational cost, and storage overhead.

A. Communication overhead

To analyze the communication overhead, we assume that the five messages (i.e., *Message 1*, *Message 2*, *Message 3*, *Message 4*, and *Message 5*) are transmitted during the registration procedure and the authentication procedure. More precisely, *Message 1* and *Message 2* are transmitted in the registration procedure and *Message 3*, *Message 4*, and *Message 5* are transmitted in the authentication procedure. *Message 1* includes the sensor's identity ID_i and *Message 2* includes f_{2i} and f_{3i}. In addition, *Message 3* includes M_1, M_2, and AID_i which are calculated as follows:

- $\mid M_1 \mid = \mid h(f_{2i}) \oplus R_1 \mid$
- $\mid M_2 \mid = \mid h(R_1) \parallel M_1 \parallel AID_i \mid$
- $\mid AID_i \mid = \mid h(R_1) \oplus ID_i \mid$

Moreover, the parameters $M_1^{'}, M_2^{'}$, and AID_j in *Message 4*, are calculated as follows:

TABLE IV
COMPUTATIONAL COST OF OUR PROPOSED MECHANISM

	Smart Sensor	Router	Authentication Server
Registration	—	—	$C_{XOR} + 2 * C_h$
Authentication	$4 * C_{XOR} + 7 * C_h + C_{ran}$	$6 * C_{XOR} + 7 * C_h + C_{ran}$	—

- $\mid M_1^{'} \mid = \mid h(ID_i) \oplus f_{1i} \mid$
- $\mid M_2^{'} \mid = \mid h(R_2 \parallel M_1^{'} \parallel AID_j) \mid$
- $\mid AID_j \mid = \mid h(ID_i) \oplus R_2 \mid$

Finally, the parameter $M_1^{''}$ in *Message 5*, is calculated as follows:

- $\mid M_1^{''} \mid = \mid SK_{ij} \oplus h(R_2) \mid$

Table III contains the setting of parameters that we have assumed for evaluating the communication overhead of the proposed mechanism. Therefore, the overall bandwidth overhead of the proposed mechanism is calculated as follows:

$$bw = \sum_{i=1}^{5} Message\ i \tag{1}$$

- *Message 1* $=\mid ID_i \mid$ = 128 bits
- *Message 2* $=\mid f_{2i} \mid + \mid f_{3i} \mid$ = 256 bits
- *Message 3* $=\mid M_1 \mid + \mid M_2 \mid + \mid f_{3i} \mid + \mid AID_i \mid$= 768 bits
- *Message 4* $=\mid M_1^{'} \mid + \mid M_2^{'} \mid + \mid AID_j \mid$= 640 bits
- *Message 5* $=\mid M_1^{''} \mid$= 128 bits

B. Computational cost

To calculate the computational cost of the proposed mechanism, we have considered the following notations: C_h denotes the cost of one-way hash function; C_{XOR} denotes the cost of XOR operation; and C_{ran} denotes the cost of generating a random number. Based on the three components (i.e, smart sensor, router, and authentication server) which are used in the proposed mechanism, the computational cost of each component is presented as follows:

1) **Smart Sensor**
   The smart sensor performs computations only in the authentication procedure. Thus, the total computational cost of this procedure is $4 * C_{XOR} + 7 * C_h + C_{ran}$.
2) **Router**
   The router performs computations only in the authentication procedures. Therefore, it needs to perform $6 * C_{XOR} + 7 * C_h + C_{ran}$ operations.
3) **Authentication Server**
   The authentication server performs $C_{XOR} + 2 * C_h$ operations to allow a smart sensor to be registered.

The proposed mechanism's computational cost is illustrated in Table IV. Due to the fact that the proposed authentication mechanism is based only on XOR operation and hash operation is efficient in terms of the computational cost.

C. Storage overhead

In the proposed mechanism, only a few parameters are needed to be stored by the smart sensor, the router, and the AS. The smart sensor stores its unique identity number ID_i and the following parameters: f_{2i}, f_{3i}, R_1, R_2, AID_i, AID_j, M_1, M_2, $M_1^{'}$, $M_2^{'}$, $M_1^{''}$, and SK_{ij}. In the other side, the router stores the pre-shared key PSK and the following parameters: f_{1i}, f_{3i}, R_1, R_2, ID_i, AID_i, AID_j, M_1, M_2, $M_1^{'}$, $M_2^{'}$, $M_1^{''}$, and SK_{ij}.

Moreover, the AS stores f_{1i}, f_{2i}, f_{3i}, and PSK. Following the setting of parameters presented in Table III, the proposed mechanism's storage cost is illustrated in Table V.

TABLE V
STORAGE COST OF OUR PROPOSED MECHANISM

	Smart Sensor	Router	Authentication Server
ID_i	✓	✓	—
AID_i	✓	✓	—
AID_j	✓	✓	—
R_1	✓	✓	—
R_2	✓	✓	—
M_1	✓	✓	—
M_2	✓	✓	—
$M_1^{'}$	✓	✓	—
$M_2^{'}$	✓	✓	—
$M_1^{''}$	✓	✓	—
SK_{ij}	✓	✓	—
f_{1i}	—	✓	✓
f_{2i}	✓	—	✓
f_{3i}	✓	✓	✓
PSK	—	✓	✓
Total cost (bits)	2176	2304	512

VI. CONCLUSION

In this paper, we have proposed a lightweight authentication mechanism, based only on hash and XOR operations, for M2M communications in Industrial IoT environment. The proposed mechanism is characterized by low computational cost, communication and storage overhead, while achieving mutual authentication, session key agreement, device's identity confidentiality, and resistance against the following attacks: replay attack, man-in-the-middle attack, impersonation attack, and modification attack. As future work, we plan to extend the proposed authentication mechanism to provide lightweight mutual authentication between sensors in Industrial IoT environment.

ACKNOWLEDGMENT

The work/A part of the work has been performed in the project Power Semiconductor and Electronics Manufacturing 4.0 (SemI40), under grant agreement No 692466. The project is co-funded by grants from Austria, Germany, Italy, France, Portugal (from the Fundação para a Ciência e Tecnologia - ECSEL/0009/2015) and - Electronic Component Systems for European Leadership Joint Undertaking (ECSEL JU).

REFERENCES

[1] S. Mumtaz, A. Alsohaily, Z. Pang, A. Rayes, K. F. Tsang, and J. Rodriguez, "Massive internet of things for industrial applications: Addressing wireless iiot connectivity challenges and ecosystem fragmentation," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 28–33, March 2017.

[2] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[3] M. Waidner and M. Kasper, "Security in industrie 4.0 - challenges and solutions for the fourth industrial revolution," in *2016 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2016, pp. 1303–1308.

[4] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, May 2014, pp. 1–4.

[5] A. W. Colombo, S. Karnouskos, O. Kaynak, Y. Shi, and S. Yin, "Industrial cyberphysical systems: A backbone of the fourth industrial revolution," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 6–16, March 2017.

[6] L. Wang, M. Törngren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *Journal of Manufacturing Systems*, vol. 37, Part 2, pp. 517 – 527, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0278612515000400

[7] M. E. Porter and J. E. Heppelmann, "How smart, connected products are transforming companies," *Harvard Business Review*, vol. 93, no. 10, pp. 96–114, 2015.

[8] L. Monostori, "Cyber-physical production systems: Roots, expectations and r&d challenges," *Procedia CIRP*, vol. 17, pp. 9 – 13, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212827114003497

[9] A. R. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial internet of things," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2015, pp. 1–6.

[10] D. Dzung, M. Naedele, T. P. V. Hoff, and M. Crevatin, "Security for industrial communication systems," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1152–1177, June 2005.

[11] L. J. Wells, J. A. Camelio, C. B. Williams, and J. White, "Cyber-physical security challenges in manufacturing systems," *Manufacturing Letters*, vol. 2, no. 2, pp. 74 – 77, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2213846314000066

[12] J. Y. Lee, W. C. Lin, and Y. H. Huang, "A lightweight authentication protocol for internet of things," in *2014 International Symposium on Next-Generation Electronics (ISNE)*, May 2014, pp. 1–2.

[13] X. Yao, X. Han, X. Du, and X. Zhou, "A lightweight multicast authentication mechanism for small scale iot applications," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3693–3701, Oct 2013.

[14] Y. Qiu and M. Ma, "A mutual authentication and key establishment scheme for m2m communication in 6lowpan networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2074–2085, Dec 2016.

[15] J. L. Hernández-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, April 2015.

[16] W. L. Chin, Y. H. Lin, and H. H. Chen, "A framework of machine-to-machine authentication in smart grid: A two-layer approach," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 102–107, December 2016.

[17] H. R. Hussen, G. A. Tizazu, M. Ting, T. Lee, Y. Choi, and K.-H. Kim, "Sakes: Secure authentication and key establishment scheme for m2m communication in the ip-based wireless sensor network (6l0wpan)," in *Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on*. IEEE, 2013, pp. 246–251.

[18] Y. Qiu and M. Ma, "A mutual authentication and key establishment scheme for m2m communication in 6lowpan networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2074–2085, 2016.

[19] A. Banks and R. Gupta, "Mqtt version 3.1. 1," *OASIS standard*, 2014.

[20] M. HiveMQ Enterprise, "Broker,"mqtt security fundamentals: Tls/ssl"."

[21] M. Singh, M. Rajan, V. Shivraj, and P. Balamuralidhar, "Secure mqtt for internet of things (iot)," in *Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on*. IEEE, 2015, pp. 746–751.

[22] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM conference on Computer and communications security*. Acm, 2006, pp. 89–98.

[23] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Security and Privacy, 2007. SP'07. IEEE Symposium on*. IEEE, 2007, pp. 321–334.

[24] R. Godfrey, D. Ingham, and R. Schloming, "Advanced message queuing protocol (amqp) websocket binding (wsb) version 1.0," 2014.

[25] K. Roebuck, *Advanced Message Queuing Protocol (AMQP): High-impact Strategies-What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Emereo Publishing, 2012.

[26] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (coap)," 2014.

[27] J. Granjal, E. Monteiro, and J. S. Silva, "Application-layer security for the wot: extending coap to support end-to-end message security for internet-integrated sensing applications," in *International Conference on Wired/Wireless Internet Communication*. Springer, 2013, pp. 140–153.

[28] S. Raza, H. Shafagh, K. Hewage, R. Hummen, and T. Voigt, "Lithe: Lightweight secure coap for the internet of things," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3711–3720, 2013.

[29] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*. IEEE, 2004, pp. 455–462.

[30] P. Saint-Andre, K. Smith, R. Tronçon, and R. Troncon, *XMPP: the definitive guide*. " O'Reilly Media, Inc.", 2009.

[31] P. Saint-Andre, "Streaming xml with jabber/xmpp," *IEEE internet computing*, vol. 9, no. 5, pp. 82–89, 2005.

[32] O. M. G. (OMG). (2015) Data distribution services specification, v1.2. [Online]. Available: http://www.omg.org/spec/DDS/1.2/

[33] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[34] L. Arnoys, "The internet of things: communicating with the cloud, the protocols, security and big data," 2015.

[35] B. Adiga, P. Balamuralidhar, M. Rajan, R. Shastry, and V. Shivraj, "An identity based encryption using elliptic curve cryptography for secure m2m communication," in *Proceedings of the First International Conference on Security of Internet of Things*. ACM, 2012, pp. 68–74.

[36] M. Klimke and J. Haid. (2016) Hardware-based secure identities for machines in smart factories. [Online]. Available: http://www.infineon.com/dgdl/Infineon-Hardware-based+Secure+Identities+for+machines+in+smart+factories-WP-v06_16-EN.pdf?fileId=5546d46254e133b401557ce235c85850

[37] J. Haid. (2016) Hardware-based solutions secure machine identities in smart factories. [Online]. Available: http://www.infineon.com/dgdl/Infineon-IoT+Security+in+Smart+Factories-ART-v01_00-EN.pdf?fileId=5546d46254e133b40154e22c8a7d0251

[38] M.-C. Chuang and J.-F. Lee, "Team: Trust-extended authentication mechanism for vehicular ad hoc networks," *IEEE systems journal*, vol. 8, no. 3, pp. 749–758, 2014.

[39] F. Wen and X. Li, "An improved dynamic id-based remote user authentication with key agreement scheme," *Computers & Electrical Engineering*, vol. 38, no. 2, pp. 381–387, 2012.

[40] H.-C. Hsiang and W.-K. Shih, "Improvement of the secure dynamic id based remote user authentication scheme for multi-server environment," *Computer Standards & Interfaces*, vol. 31, no. 6, pp. 1118–1123, 2009.

[41] Y.-P. Liao and S.-S. Wang, "A secure dynamic id based remote user authentication scheme for multi-server environment," *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 24–29, 2009.