Breaking a Lightweight M2M Authentication Protocol for Communications in IIoT Environment

Seyed Farhad Aghili and Hamid Mala

Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran

{sf.aghili@eng, h.mala@eng}.ui.ac.ir

Abstract. The concept of the Industrial Internet of Things (IIoT) can be defined as the integration of smart sensor networks and the Internet of Things (IoT). This technology can be employed in various industries such as agriculture, food processing industry, environmental monitoring, security surveillance, and so on. Generally, a smart sensor is a resource-constrained device which is responsible for gathering data from the monitored area. Machine-to-Machine (M2M) communication is one of the most important technologies to exchange information between entities in industrial areas. However, due to the insecure wireless communication channel and the smart sensor's limitations, security and privacy concerns are the important challenges in IIoT environments. The goal of this paper is to address the security flaws of a recent M2M authentication protocol proposed for employing in IIoT including DoS, router impersonation and smart sensor traceability attacks. Moreover, we showed that an untrusted smart sensor can obtain the secret key of the router and the session key which another sensor establishes with the target router.

keywords: M2M communications, IIoT, Authentication, DoS Attack, Traceability Attack, Impersonation attack, Disclosure Attack.

1 Introduction

Machine-to-Machine (M2M) is a crucial technology for realization of Industrial Internet of Things (IIoT). In M2M technology, all devices, machines, and equipments can communicate between each other through wireless links. M2M technology can provide the exchange of data between typically resource-constrained devices without human intervention. M2M networks consist of a number of smart sensors, routers and an authentication server (*AS*) responsible for performing the registration procedure and generating the secure pre-shared

keys for entities. The important concept of M2M is to enable real-time data communications between the device equipped with smart sensor and the central management applications to collect the vital data transfered from the remote devices for their users [9, 10].

Since M2M technology lies within embedded cellular via 3GPP technologies such as GSM/GPRS, UMTS/HSPA(+), and LTE networks, it has become mobile and smarter than before. However, similar to wireless sensor networks (WSNs), 3GPP cellular networks are vulnerable to some well known information security threats [7]. So, similar to WSNs, there is a need for assurance regarding the confidentiality, integrity of the transfered data and robustness against attacks from external entities. These security requirements can be fulfilled by designing the secure and efficient communication schemes.

To ensure secure communication between M2M devices, authentication and key agreement (AKA) schemes have been proposed. These schemes not only should be able to securely accomplish mutual authentication and session key establishment but also should meet the high efficiency.

In M2M technology employed in IIoT, the system model is composed of three primary entities: a smart sensor, router and authentication server (*AS*), which is described in Fig. 1.

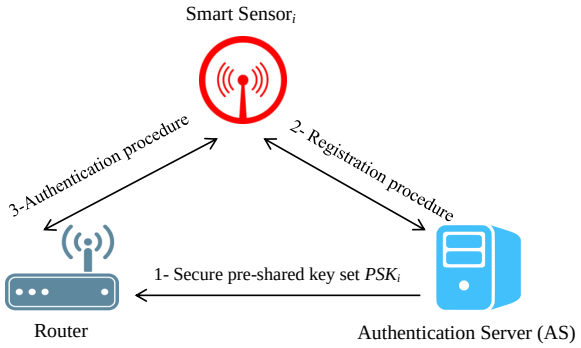

Fig. 1. System model of M2M technology employed in IIoT

1.1 Threat Model

Our threat model follows the Dolev-Yao model [4]. In this model, an adversary can intercept, modify, delete or change the contents of the transmitted messages through the insecure channel between any two entities.

2 Related work

Several authentication protocols for M2M communications has been published since 2012. In 2012, Chen *et al.* [3] proposed a group authentication and key agreement protocol called G-AKA. In their scheme a serving network (SN) can authenticate a group of mobile stations (MSs) with assistance of home network (HN). Chen *et al.* claimed that their proposed scheme is secure and can satisfy all security requirements. However, their protocol is vulnerable to man-in-the-middle, DoS and redirection attacks [6]. In 2013, Lai et al. [6] proposed a group authentication and key agreement protocol, called SE-AKA. Their scheme is efficient and guarantees privacy and forward/backward key secrecy. Later in [1], the authors proposed a group authentication and key agreement (GROUP-AKA) protocol designed for M2M communication. In this protocol, the authors successfully prevent the DoS attack, but it still suffers from the privacy preservation problem. In addition, this protocol employs asymmetric cryptosystem based operations with high computation overhead.

In [8] the authors presented another group authentication and key agreement protocol for M2M devices in 3GPP networks called GLARM. This mutual authentication and secure key agreement protocol has been proposed for recourse-constrained devices and has two main phases: (1) Initialization phase and (2) Group authentication and key agreement phase and proposed for resource-constrained devices. In 2017, Chen et al. [2] proposed an authentication protocol for the IoT, called S2M but the authors did not analyze the performance of the methods in terms of privacy preservation, especially in comparison to the scheme presented in [8].

Recently, Esfahani et al. [5] proposed a lightweight M2M authentication protocol to ensure secure integration of Industrial Internet of Things (IIoT) solutions. Specifically, in their work a machine equipped with a smart sensor which is authenticated by a network element equipped with a Trusted Platform Module (TPM). Esfahani et al. assumed that their protocol is secure and efficient. However, in this paper we show that their scheme is vulnerable against DoS and the router impersonation attacks and also we demonstrate that the adversary can trace the smart sensor by eavesdropping only one session. Besides, we show how an untrusted smart sensor can obtain the secret key of the router, PSK, and the session key which another sensor establishes with the target router.

Paper organization The remainder of the article is organized as follows. The review of the Esfahani *et al.*'s scheme is presented in Section 3. In Section 4, we discuss serious weaknesses of this scheme. Finally, we conclude the paper in Section 5.

3 Review of Esfahani *et al.*'s scheme

In this section, we review Esfahani *et al.*'s authentication protocol [5], which is composed of three phases, i.e., initialization, registration and authentication, and key agreement phases.

The notations used in this paper are listed in Table 1.

Table 1. Notations

Notation	Description
AS	The authentication server
ID_i	Identity of i-th smart sensor
AID_i	The alias of the i-th identity
x	Secret key of the AS
PSK_j	A secure pre-shared key between the AS and the j-th router
SK	Session key
R_1, R_2	Random number generated by a Pseudo random Number Generator (PRNG)
$h(\cdot)$	One-way hash function
$\oplus$	XOR operation
$\|$	Concatenation operation

3.1 Initialization phase

In this phase, the authentication server AS initially chooses a long-term secret key x and generates the secure pre-shared key set PSK_j, $j = 1, ..., n$ and sends PSK_j to the j-th router.

3.2 Registration phase

In this phase, a smart sensor performs the following steps with AS through a secure channel (Fig. 2).

Step 1. The smart sensor chooses its identity ID_i and submits it to the AS.

Step 2. If the AS does not find ID_i in the database, it calculates $f_{1i} = h(ID_i \| x)$, $f_{2i} = h(f_{1i})$ and $f_{3i} = PSK_j \oplus f_{1i}$ and delivers $\langle f_{2i}, f_{3i} \rangle$ to the smart sensor.

Step 3. After receiving the message, the smart sensor writes $\langle f_{2i}, f_{3i} \rangle$ to the Secure Element SE of the sensor.

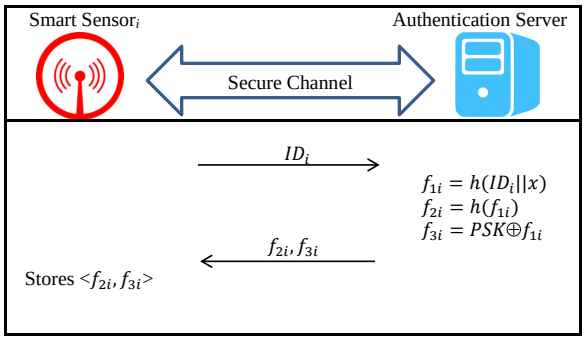

Fig. 2. Registration phase of Esfahani *et al.*'s scheme [5]

3.3 Authentication and Session Key Agreement Phase

In this phase, a smart sensor and a router are authenticated each other and a session key is established between them, as shown in Fig. 3.

Step 1. The smart sensor starts the protocol by generating a random number R_1, computing $M_1 = h(f_{2i}) \oplus R_1$, $AID_i = h(R_1) \oplus ID_i$, $M_2 = h(R_1 \| M_1 \| AID_i)$ and sending $\langle M_1, M_2, f_{3i}, AID_i \rangle$ to the router.

Step 2. After receiving the message $\langle M_1, M_2, f_{3i}, AID_i \rangle$, the router extracts $f_{1i} = f_{3i} \oplus PSK_j$, $R_1 = M_1 \oplus h(f_{2i})$ and $ID_i = AID_i \oplus h(R_1)$, and checks whether $h(R_1 \| M_1 \| AID_i) = M_2$ holds. If it does not hold, the router terminates the connection; otherwise, it generates the random number R_2, calculates $AID_j = R_2 \oplus h(ID_i)$, $M_1' = f_{1i} \oplus h(ID_i)$, $M_2' = h(M_1' \| AID_j \| R_2)$ and then computes $SK_{ij} = h(R_1 \| R_2)$ as a session key and then sends the tuple $\langle M_1', M_2', AID_j \rangle$ to the smart sensor.

Step 3. Upon receiving the message $\langle M_1', M_2', AID_j \rangle$, the smart sensor computes $R_2 = AID_j \oplus h(ID_i)$ and checks whether $h(M_1' \| AID_j \| R_2) = M_2'$ holds. If it does not hold, the smart sensor terminates the session; otherwise, it computes the session key $SK_{ij} = h(R_1 \| R_2)$ and $M_1'' = SK_{ij} \oplus h(R_2)$. Finally, the smart sensor sends $\langle M_1'' \rangle$ to the router.

Step 4. To check the validity of the received $\langle M_1'' \rangle$, the router checks whether $M_1'' \oplus SK_{ij} = h(R_2)$ holds. If it does not hold, the router terminates the session; otherwise, it means the entities mutually authenticate each other and establish the session key $SK_{ij} = h(R_1 \| R_2)$.

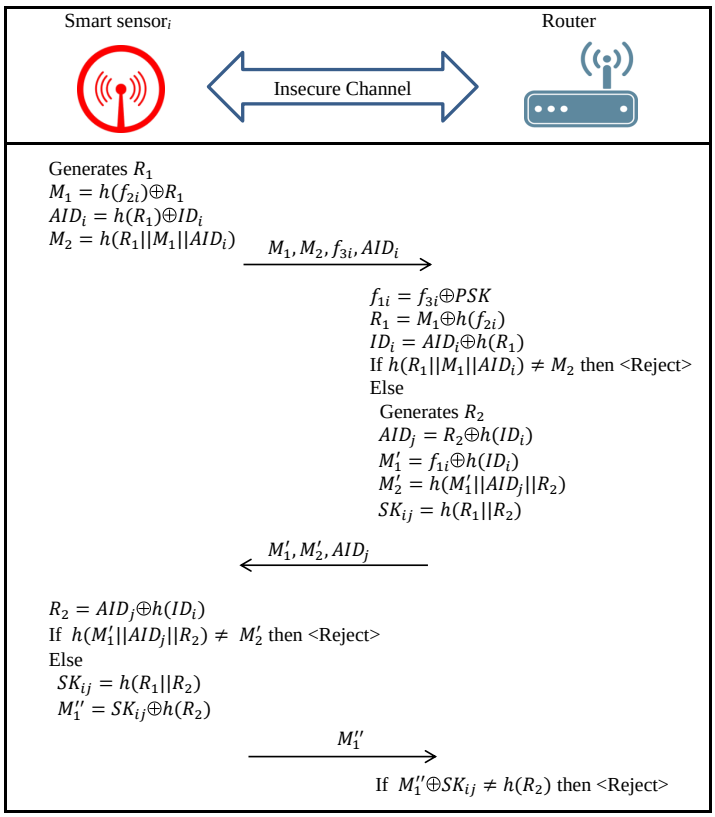

Fig. 3. Authentication and key agreement phase of Esfahani *et al.*'s scheme [5]

4 Security Analysis of the Esfahani *et al.*'s Protocol

In Esfahani *et al.*'s Protocol, the authors assumed that the adversary A is not able to execute replay attack using the transmitted messages. Moreover, they assumed that the adversary cannot impersonate the router by accessing to the old login eavesdropped messages and also they assumed that the legitimate smart sensor will be identified by the router before the calculation of session key.

In this section, we demonstrate DoS attack and the router impersonation attack and also we show that the adversary can trace the smart sensor by eavesdropping only one session. Besides, we show how an untrusted smart sensor can obtain the secret key of the router, PSK, and the session key which another sensor has previously established with the target router.

Router impersonation attack The purpose of adversary in this attack is to cheat the smart sensor to establish a new session key with her. In this attack it is sufficient for the adversary to eavesdrop the tuple messages $\langle M_1', M_2', AID_j \rangle$ and performs the following steps.

Step 1. The smart sensor starts the protocol by generating the random number R_1, computing $M_1 = h(f_{2i}) \oplus R_1$, $AID_i = h(R_1) \oplus ID_i$, $M_2 = h(R_1 \| M_1 \| AID_i)$ and sending $\langle M_1, M_2, f_{3i}, AID_i \rangle$ to the router which is the adversary.

Step 2. The adversary sends the previously eavesdropped messages $\langle M_1', M_2', AID_j \rangle$ to the smart sensor.

Step 3. The smart sensor computes $R_2 = AID_j \oplus h(ID_i)$. Hence $h(M_1' \| AID_j \| R_2) = M_2'$ holds, the smart sensor computes the session key $SK_{ij} = h(R_1 \| R_2)$ and $M_1'' = SK_{ij} \oplus h(R_2)$. Finally, the smart sensor sends $\langle M_1'' \rangle$ to the adversary.

Hence, by following this attack the adversary is authenticated as a legitimate router, so the smart sensor believes that the router holds the legitimate session key $SK_{ij} = h(R_1 \| R_2)$. The main cause of this attack is that the parameters used in messages $\langle M_1', M_2', AID_j \rangle$ are independent of time and smart sensor's random number.

DoS attack In the Esfahani *et al.* protocol, the adversary starts the attack by eavesdropping the tuple messages $\langle M_1, M_2, f_{3i}, AID_i \rangle$ and simulates the smart sensor by replaying them. The main cause of this attack is that the parameters used in messages $\langle M_1, M_2, f_{3i}, AID_i \rangle$ are independent of time and lack any random generated by the router. The adversary executes the attack as follows.

Step 1. The adversary sends eavesdropped messages $\langle M_1, M_2, f_{3i}, AID_i \rangle$ to the router;

Step 2. The router extracts $f_{1i} = f_{3i} \oplus PSK_j$, $R_1 = M_1 \oplus h(f_{2i})$ and $ID_i = AID_i \oplus h(R_1)$. Hence $h(R_1 \| M_1 \| AID_i) = M_2$ holds, the router generates the random number R_2, calculates $AID_j = R_2 \oplus h(ID_i)$, $M_1' = f_{1i} \oplus h(ID_i)$, $M_2' = h(M_1' \| AID_j \| R_2)$ and then computes $SK_{ij} = h(R_1 \| R_2)$ as the session key and then sends the tuple $\langle M_1', M_2', AID_j \rangle$ to the smart sensor which is the adversary.

Although the authentication fails in the last step of the protocol, but the adversary has imposed the whole computations of a session to the router. So, this attack is regarded as a DoS attack.

Smart sensor traceability attack In this attack, the adversary tries to find the link between two sessions of the protocol. In Esfahani *et al.* protocol the attacker eavesdrops the message $\langle M_1, M_2, f_{3i}, AID_i \rangle$ of each session and because f_{3i} is unchanged, the attacker can use it to distinguish and track the target smart sensor.

Secret disclosure attack In the Esfahani *et al.* protocol, *j-th* router pre-shares a unique key, PSK_j, with AS. So, an honest but curious smart sensor, say $sensor_i$ with identity ID_i, can obtain PSK_j as follows.

- Uses message $M_1' = f_{1i} \oplus h(ID_i)$ and $h(ID_i)$ and obtains f_{1i} from equation $f_{1i} = M_1' \oplus h(ID_i)$;
- Computes the *j-th* router's secret key PSK_j using the equation $PSK_j = f_{3i} \oplus f_{1i}$.

Thus, Esfahani *et al.*'s protocol is vulnerable against secret disclosure attack.

Session key disclosure attack As mentioned above, the honest but curious smart sensor can obtain PSK_j. By employing PSK_j, such a smart sensor can obtain another *l-th* smart sensor's session key SK_{lj} which is established with the same router, say the *j-th* router, as bellow.

- The honest but curious *i-th* smart sensor uses the message $f_{3l} = PSK_j \oplus f_{1l}$ transfered by *l-th* smart sensor and PSK_j to obtain f_{1l} from equation $f_{1l} = f_{3l} \oplus PSK_j$;
- It then uses the message $M_1' = f_{1l} \oplus h(ID_l)$ and f_{1l} and obtains $h(ID_l)$ from equation $h(ID_l) = M_1' \oplus f_{1l}$;
- After that, it employs the message $AID_j = R_2 \oplus h(ID_l)$ and $h(ID_l)$ and obtains R_2 from equation $R_2 = AID_j \oplus h(ID_l)$;
- Finally, it computes $h(R_2)$ and uses the message $M_1'' = SK_{lj} \oplus h(R_2)$ to obtain SK_{lj} from equation $SK_{lj} = M_1'' \oplus h(R_2)$;

Thus, Esfahani *et al.*'s protocol is vulnerable to session key disclosure attack.

5 Conclusion

In this paper we considered the security of the Esfahani *et al.* authentication M2M protocol proposed for communication in IIoT environment [5]. We affirmed that Esfahani *et al.*'s protocol is vulnerable to DoS and the router impersonation attacks. We also showed how an adversary can trace the smart sensor by eavesdropping only one session. Moreover, we showed that an untrusted smart sensor can obtain the secret key of the router *PSK* and the session key which another sensor establishes with the router.

References

1. J. Cao, M. Ma, and H. Li. Gbaam: group-based access authentication for mtc in lte networks. *Security and Communication Networks*, 8(17):3282–3299, 2015.
2. D. Chen, N. Zhang, Z. Qin, X. Mao, Z. Qin, X. Shen, and X.-Y. Li. S2m: A lightweight acoustic fingerprints-based wireless device authentication protocol. *IEEE Internet of Things Journal*, 4(1):88–100, 2017.
3. Y.-W. Chen, J.-T. Wang, K.-H. Chi, and C.-C. Tseng. Group-based authentication and key agreement. *Wireless Personal Communications*, 62(4):965–979, 2012.
4. D. Dolev and A. Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 1983.
5. A. Esfahani, G. Mantas, R. Matischek, F. B. Saghezchi, J. Rodriguez, A. Bicaku, S. Maksuti, M. Tauber, C. Schmittner, and J. Bastos. A lightweight authentication mechanism for m2m communications in industrial iot environment. *IEEE Internet of Things Journal*, 2017.
6. C. Lai, H. Li, R. Lu, and X. S. Shen. Se-aka: A secure and efficient group authentication and key agreement protocol for lte networks. *Computer Networks*, 57(17):3492–3510, 2013.
7. C. Lai, H. Li, Y. Zhang, and J. Cao. Security issues on machine to machine communications. *KSII Transactions on Internet & Information Systems*, 6(2), 2012.
8. C. Lai, R. Lu, D. Zheng, H. Li, and X. S. Shen. Glarm: Group-based lightweight authentication scheme for resource-constrained machine to machine communications. *Computer Networks*, 99:66–81, 2016.
9. G. Tuna, D. G. Kogias, V. C. Gungor, C. Gezer, E. Taşkın, and E. Ayday. A survey on information security threats and solutions for machine to machine (m2m) communications. *Journal of Parallel and Distributed Computing*, 109:142–154, 2017.
10. Y. Zhang, J. Chen, H. Li, W. Zhang, J. Cao, and C. Lai. Dynamic group based authentication protocol for machine type communications. In *Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on*, pages 334–341. IEEE, 2012.