Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Notes on "Secure authentication scheme for IoT and cloud servers"

Chin-Chen Chang [a,*], Hsiao-Ling Wu [a], Chin-Yu Sun [b]

[a] Department of Information Engineering and Computer Science, Feng Chia University, Taichung, 407, Taiwan

[b] Department of Computer Science, National Tsing-Hua University, Hsinchu, 30013, Taiwan

ARTICLE INFO

Article history:
Received 16 October 2015
Received in revised form 1 December 2015
Accepted 1 December 2015
Available online xxxx

Keywords:
Internet of Things
Cryptanalysis
Authentication
Elliptic curve cryptography

ABSTRACT

In 2015, Kalra and Sood proposed an authentication scheme for the IoT and a Cloud server. In their paper, the authors pointed out that the embedded devices in both the IoT and the Cloud server cannot support high computational and storage abilities. For these reasons, Kalra and Sood used ECC to design a light-weight scheme. Unfortunately, we found two weaknesses in it, i.e., the failure of mutual authentication and a mistiness of the session key. In this research, we first demonstrate the weaknesses, and then we improve their scheme to make the original scheme-more complete and more secure.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Due to the rapid development of the Internet of Things (IoT) [1], people have started connecting their many personal embedded devices to the Internet to transmit information. For example, a user can use his/her mobile phone to control a series of devices in his/her office remotely on his/her way to the company, i.e., computers, lamps, or air conditioners. Although IoT makes for more convenience, it still limited its power and the computation ability of the embedded device. For this reason, people will combine it with the Cloud server. A Cloud server is a platform that has more resources and abilities available. Cooperating with the Cloud server, the device can then rely on the computation ability of the cloud server and the device more suitable for practical implementation in IoT. As a result, at the same time, more and more people can effortlessly acquire different types of IoT service using their embedded devices. However, both authentication and the created session key between the user and the server are two problems in IoT.

In 2014, Liao and Hsiao [2] proposed an ECC-based authentication scheme for IoT. Liao and Hsiao combined their scheme with a secure ID-verifier transfer protocol and focused on the security available between the radio frequency identification (RFID) tag and a server. In 2014, Turkanović et al. [3] proposed a user authentication and key agreement scheme using a Smartcard for IoT. After registering in Turkanović et al.'s scheme, a legal user would get one Smartcard for helping him/her to log into the server. However, Farash et al. [4] pointed out the Turkanović et al.'s scheme had security weaknesses and proposed a new scheme to improve Turkanović et al.'s scheme the same year.

In 2015, Kalra and Sood [5] proposed an authentication and key agreement scheme for Internet of Things and the Cloud server based on "Elliptic Curve Cryptography" [5–7]. The properties of ECC make Kalra and Sood's scheme both efficient and secure. In their scheme, a user and a server can first authenticate each other and then negotiate one short-term session

[*] Correspondence to: Department of Information Engineering and Computer Science, Feng Chia University, No. 100 Wenhwa Rd., Seatwen, Taichung 407, Taiwan. Fax: +886 4 27066495.

E-mail addresses: alan3c@gmail.com (C.-C. Chang), wuhsiaoling590@gmail.com (H.-L. Wu), sun.chin.yu@gmail.com (C.-Y. Sun).

http://dx.doi.org/10.1016/j.pmcj.2015.12.003
1574-1192/© 2016 Elsevier B.V. All rights reserved.

key to use. In spite of Kalra and Sood proofs indicating their scheme achieves mutual authentication and provides essential security requirements by security analysis, we found two weaknesses in it: (1) a failure of mutual authentication and (2) mistiness of the session key. This shortcoming is demonstrated and analyzed in detail in Section 2. In Section 3, an improved scheme is then proposed and discussed. Finally, our conclusions are offered in Section 4.

2. Review and analysis of Kalra and Sood's scheme

In this section, we first introduce each procedure in [5]. Kalra and Sood's scheme includes three phases, (1) registration, (2) pre-computation and login, and (3) authentication. The processes for each phase are described in Section 2.1, 2.2, and 2.3, respectively. We also point out their scheme cannot achieve mutual authentication and session key agreement. This analysis is shown in Sections 2.4 and 2.5.

2.1. Registration phase

When an embedded device D_i wants to obtain service from server S, D_i needs to register with S in this phase. Therefore, D_i first sends the unique identity ID_i to S. Once S receives the unique identity of the embedded device D_i, S generates a password P_i and a random number R_i for the D_i. Subsequently, S computes

$$\begin{aligned} &CK = H(R_i \parallel X \parallel EXP\_Time \parallel ID_i),\\ &CK' = CK \times G,\\ &T_i = R_i \oplus H(X),\\ &A_i = H(R_i \oplus H(X) \oplus P_i \oplus CK'),\\ &\text{and} \quad A_i' = A_i \times G, \end{aligned} \tag{1}$$

where X is the private key of Server S and EXP_Time is the expiration time. Finally, S sends CK' back to D_i, and stores $\{A_i', T_i, ID_i, EXP\_Time\}$ in its database.

2.2. Pre-computation and login phase

After the registration phase, the embedded device D_i obtains authentication token CK'. D_i can use this authentication token to compute the authentication required message. Firstly, D_i chooses a random number N_1, and uses CK' and N_1 to compute

$$\begin{aligned} &P_1 = N_1 \times G,\\ &\text{and} \quad P_2 = H(N_1 \times CK'). \end{aligned} \tag{2}$$

Finally, D_i sends the authentication required message $\{ID_i, P_1, P_2\}$ to S for the next phase.

2.3. Authentication phase

Once S receives $\{ID_i, P_1, P_2\}$, S uses ID_i to search itself database. If S finds the record $\{A_i', T_i, ID_i, EXP\_Time\}$, S can use T_i, EXP_Time, and private key X to compute

$$\begin{aligned} &R_i = T_i \bigoplus H(X),\\ &CK = H(R_i \| X \| EXP\_Time \parallel ID_i),\\ &\text{and} \quad P_2' = H(P_1 \times CK). \end{aligned} \tag{3}$$

Then, S checks $P_2'? = P_2$. If the value of P_2' is not equal to that of P_2, it means the embedded device D_i is not a legal user, and the authentication process is terminated. Otherwise, S will choose a random number N_2 to compute

$$P_3 = N_2 \times G \quad \text{and} \quad P_4 = N_2 \times A_i', \tag{4}$$

and then S sends a response message $\{T_i, P_3, P_4\}$ to D_i. When D_i receives $\{T_i, P_3, P_4\}$, it computes

$$A_i = H(T_i \oplus P_i \oplus CK') \quad \text{and} \quad P_4' = P_3 \times A_i. \tag{5}$$

Subsequently, D_i checks $P_4'? = P_4$. If the value of P_4' is equal to that of P_4, then S is a legal server. Therefore, D_i continues computing

$$V_i = H((N_1 \times CK') \parallel P_4'), \tag{6}$$

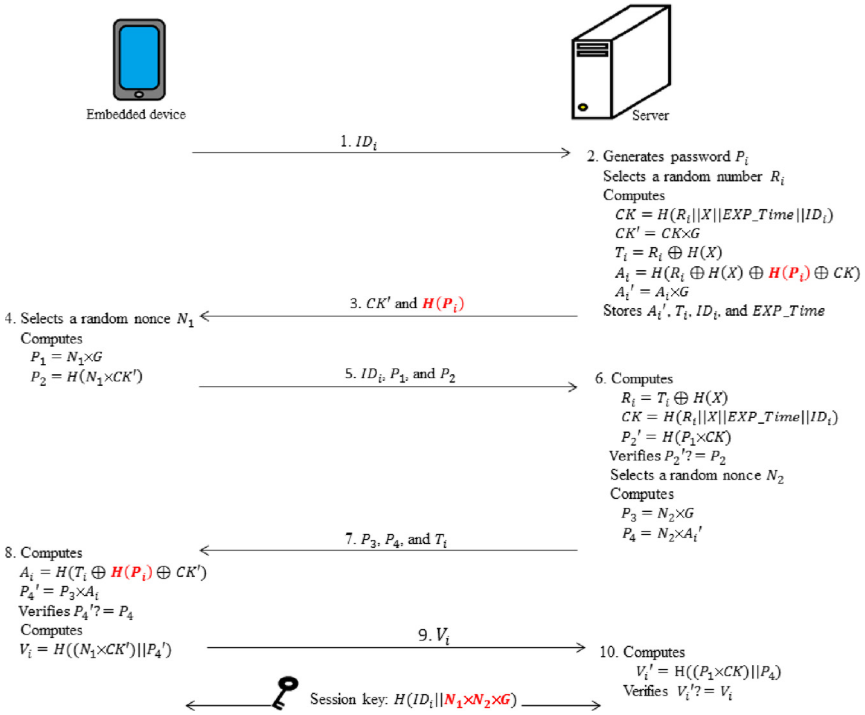
Embedded device
Server
1. ID_i
2. Generates password P_i
Selects a random number R_i
Computes
$CK = H(R_i||X||EXP\_Time||ID_i)$
$CK' = CK \times G$
$T_i = R_i \oplus H(X)$
$A_i = H(R_i \oplus H(X) \oplus H(P_i) \oplus CK)$
$A_i' = A_i \times G$
Stores A_i', T_i, ID_i, and EXP_Time
3. CK' and $H(P_i)$
4. Selects a random nonce N_1
Computes
$P_1 = N_1 \times G$
$P_2 = H(N_1 \times CK')$
5. ID_i, P_1, and P_2
6. Computes
$R_i = T_i \oplus H(X)$
$CK = H(R_i||X||EXP\_Time||ID_i)$
$P_2' = H(P_1 \times CK)$
Verifies $P_2'? = P_2$
Selects a random nonce N_2
Computes
$P_3 = N_2 \times G$
$P_4 = N_2 \times A_i'$
7. P_3, P_4, and T_i
8. Computes
$A_i = H(T_i \oplus H(P_i) \oplus CK')$
$P_4' = P_3 \times A_i$
Verifies $P_4'? = P_4$
Computes
$V_i = H((N_1 \times CK')||P_4')$
9. V_i
10. Computes
$V_i' = H((P_1 \times CK)||P_4)$
Verifies $V_i'? = V_i$
Session key: $H(ID_i||N_1 \times N_2 \times G)$

Fig. 1. The proposed scheme.

and sending V_i to S. While S receives V_i, it will compute

$$V_i' = H((P_1 \times CK) \parallel P_4), \tag{7}$$

and check $V_i'? = V_i$. Finally, the embedded device D_i will share a session key SK with the server S. The session key is computed using the following formula.

$$SK = H(X \parallel ID_i \parallel N_1 \parallel N_2). \tag{8}$$

2.4. Failure of mutual authentication

In Kalra and Sood's scheme, the authors claimed it could achieve mutual authentication by computing Formulas (4) and (5) and checking $P_2'? = P_2$ and $P_4'? = P_4$. If the server and the embedded device are legal, the equations are established, i.e., $P_2' = P_2$ and $P_4' = P_4$. The inference processes are shown as follows:

$$\begin{aligned} P_2' &= H(P_1 \times CK) = H(N_1 \times G \times CK) = H(N_1 \times CK') = P_2, \\ P_4' &= P_3 \times A_i = N_2 \times G \times A_i = N_2 \times A_i' = P_4. \end{aligned} \tag{9}$$

However, an embedded device cannot compute A_i. We know $A_i = H(T_i \oplus P_i \oplus CK')$, where T_i and CK' are delivered by the server in the authentication phase and the registration phase, respectively. The server generates the password P_i for the embedded device, but never delivers this value to that embedded device. The embedded device thus cannot compute a correct P_4' to pass verification even when the embedded device is legal. Therefore, Kalra and Sood's scheme cannot achieve mutual authentication.

2.5. Mistiness of session key

After the authentication phase, the server and the embedded device can share a session key to encrypt all the subsequent messages. The session key is computed as $SK = H(X \parallel ID_i \parallel N_1 \parallel N_2)$. However, the server and the embedded device cannot correctly compute SK. Below, we analyze how the session key is computed on both sides.

For the server, that server can receive $\{ID_i, P_1, P_2\}$ during pre-computation and login phase, where $P_1 = N_1 \times G$ and $P_2 = H(N_1 \times CK')$. According to elliptic curve discrete logarithm problem (ECDLP) [5] and the definition of one-way hash function [8], the server is computationally infeasible to obtain N_1. Hence, the server cannot compute SK.

Please cite this article in press as: C.-C. Chang, et al., Notes on "Secure authentication scheme for IoT and cloud servers", Pervasive and Mobile Computing (2016), http://dx.doi.org/10.1016/j.pmcj.2015.12.003

For the embedded device, it can receive $\{T_i, P_3, P_4\}$, where $P_3 = N_2 \times G$ and $P_4 = N_2 \times A'_i$. According to same reason, embedded device is computationally infeasible to obtain N_2. Furthermore, embedded device cannot obtain the private key of server, X. Hence, the embedded device cannot compute SK.

As mentioned above, Kalra and Sood's scheme cannot achieve session key agreement.

3. The proposed scheme

To overcome the weaknesses discussed in Section 2, we propose a simple modified scheme that constructs three phases, (1) the registration phase, (2) the pre-computation and login phase, and (3) the authentication phase. Here, we omit the complicated descriptions of each phase because the basis of our scheme is based on [5]. However, we still do show the details of each phase in Fig. 1. Further, we discuss the differences between Kalra and Sood's scheme and our proposed scheme.

As shown in Fig. 1, the registration phase consists of Steps 1–3. In this phase, we only modify the parameter P_i to $H(P_i)$ in A_i of Step 2. Additionally, the server needs to send $H(P_i)$ to the user. Step 4 in Fig. 1 is the pre-computation and login phase of our proposed scheme, and it is the same as the pre-computation and login phase in Kalra and Sood's scheme. Next, Steps 5–10 are used for the authentication phase in our proposed scheme. In this authentication phase, we only modify the parameter P_i to $H(P_i)$ in A_i of Step 8, i.e., $A_i = H(T_i \oplus H(P_i) \oplus CK')$. After the server and the embedded device authenticate each other, they can negotiate one short-term session as $H(ID_i \parallel N_1 \times N_2 \times G)$.

Theorem 1. *Our improved scheme has the property of mutual authentication.*

Correctness. In Kalra and Sood's scheme, an embedded device cannot successfully compute A_i resulting in a failure of mutual authentication. However, in our improved scheme, the embedded device can receive $H(P_i)$, CK', and T_i from Steps 3 and 7. Then, the embedded device can successfully compute A_i and verify that the equality $P'_4? = P_4$ holds or not. Hence, our improved scheme can achieve mutual authentication.

Theorem 2. *Our improved scheme has the property of session key agreement.*

Correctness. It is obvious that the session key can be successfully computed if the server and the embedded device are legal. They can thus negotiate one short-term session key by computing $H(ID_i \parallel P_1 \times N_2)$ and $H(ID_i \parallel P_3 \times N_1)$, respectively. Thus, our improved scheme can achieve session key agreement.

4. Conclusions

In this paper, we analyze the IoT authentication and key agreement scheme proposed by Kalra and Sood. Although their scheme uses elliptic curve cryptography to enhance the security, it still suffers from two security problems, i.e., the failure of mutual authentication and a mistiness of the session key. We provide a simple modified scheme to enhance Kalra and Sood's scheme. Applying Theorems 1 and 2, our improved scheme not only solves two security problems in Kalra and Sood's scheme, but also inherits the advantages of Kalra and Sood's scheme.

References

[1] Luigi Atzori, Antonio Iera, Giacomo Morabito, The Internet of things: A survey, Comput. Netw. 54 (15) (2010) 2787–2805.

[2] Yi-Pin Liao, Chih-Ming Hsiao, A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol, Ad Hoc Networks 18 (2014) 133–146.

[3] Muhamed Turkanović, Boštjan Brumen, Marko Hölbl, A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of things notion, Ad Hoc Networks 20 (2014) 96–112.

[4] Mohammad Sabzinejad Farash, Muhamed Turkanović, Saru Kumari, Marko Hölbl, An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of things environment, Ad Hoc Networks 36 (2016) 152–176.

[5] Sheetal Kalra, Sandeep K. Sood, Secure authentication scheme for IoT and cloud servers, Pervasive Mob. Comput. 24 (2015) 210–223.

[6] Victor S. Miller, Use of elliptic curves in cryptography, in: Proceedings of Advances in Cryptology—CRYPTO'85, California, U.S.A., August 1985, Vol. 218, pp. 417–426.

[7] Neal Koblitz, Elliptic curve cryptosystems, Math. Comp. 48 (177) (1987) 203–209.

[8] Alfred J. Menezes, Scott A. Vanstone, Paul C. Van Oorschot, Handbook of Applied Cryptography, CRC Press, USA, 1996, pp. 321–376.

Please cite this article in press as: C.-C. Chang, et al., Notes on "Secure authentication scheme for IoT and cloud servers", Pervasive and Mobile Computing (2016), http://dx.doi.org/10.1016/j.pmcj.2015.12.003