The Journal of Supercomputing (2021) 77:13897–13910
https://doi.org/10.1007/s11227-021-03836-y

Secure authenticated key exchange for WSNs in IoT applications

Mingping Qi[1] · Jianhua Chen[2]

Accepted: 20 April 2021 / Published online: 5 May 2021

Abstract

Wireless sensor networks (WSNs) are the key technological building block of Internet of Things (IoT), by which remote users can access the real-time data from the sensor nodes. Due to the openness and mobility of such network, it is essential to establish secure links for end-to-end communication with proper authentication. Although amount of research works in this area have been made, so far designing a secure and efficient authenticated key exchange (AKE) protocol for this setting is still an open topic. So, the authors in this paper design a two-factor AKE protocol using elliptic curve cryptography (ECC) for WSNs in the context of IoT, allowing the end users and sensor nodes to exchange information directly after a secure link is established with the help of the corresponding gateway node. The heuristic security analysis has shown that the new protocol can provide various expected security attributes and resist various known attacks. Moreover, the performance study states that the new protocol is efficient enough and has certain efficiency advantages in the aspects of computation and communication costs compared with the same type of ECC-based AKE protocols for IoT applications.

Keywords Wireless sensor networks · IoT · Authentication · Two-factor · Elliptic curve cryptography

✉ Mingping Qi
mpqi_math@163.com

[1] School of Cybersecurity, Northwestern Polytechnical University, Xi'an 710072, People's Republic of China

[2] School of Mathematics and Statistics, Wuhan University, Wuhan 430072, People's Republic of China

1 Introduction

Internet of Things (IoT) [3, 11] as the future evolution of the Internet, is composed of various interconnect smart devices, and remote users can exchange and gain data from these IoT devices over the Internet. In the IoT application settings, wireless sensor networks (WSNs) are the key technological building blocks, and often exist as a local centralized approach to establish a hierarchical IoT network. In general, the sensor node (SN), gateway node (GWN) and the end user are the main components of the WSN, where the sensor nodes are connected to the Internet via their nearby gateway node GWN which has enough computation, communication and storage resources, and the end users can access the real-time data from the sensor nodes directly with the help of GWN. To access the data from the sensor nodes, the request messages are generally first transmitted from the remote user to the GWN and then they are forwarded to the corresponding sensor nodes.

IoT-combined-WSNs user-cases can be applied in many fields, such as smart city [25], health care [5] and industrial communication [21]. As examples, a doctor outside the hospital can provide diagnostic opinion by remotely accessing the sensor nodes installed in monitoring the health condition of patients; in auto production automation monitoring applications, WSNs can be deployed inside the factory to obtain kinds of raw data in the mechanical process and further to identify machine abnormalities and create safety alarms. However, the openness of the network connection between the end user and sensor nodes has made the communications between them susceptible to various attacks such as the impersonation attack, man-in-the-middle attack, and replay attack. Considering the sensitive of the data collected by the sensor nodes, it is essential to design a secure mechanism for authenticating the identity of connecting devices and establishing secure communication links between the end users and sensor nodes. Authenticated key exchange (AKE) protocols exactly play a key role in protecting the networks, especially, the two-factor AKE schemes can conveniently provide strong security assurance for the communication links between the end users and sensor nodes. Two-factor AKE schemes typically use password and smart card as the two authentication factors to authenticate the end users, and many relevant schemes have been presented in literatures for WSNs in IoT applications, e.g., [1, 12, 14–16, 20, 24], etc. As usual, an adequate AKE protocol for securing the WSNs should achieve the following security requirements: (i) mutual authentication and session key agreement, (ii) privacy-preserving, (iii) session key forward secrecy, (iv) resistance to various known attacks, e.g., replay attack, impersonation attack, password guessing attack, etc. Besides, the AKE protocols should be as efficient as possible.

Although there have been many AKE protocols presented for WSNs in IoT applications, some problems are still within these relevant schemes including but not limited to (i) some of them need to rely on PKI [4] certificate such as the recently presented scheme [15], making them possibly unavailable to some resource-constraint environments; (ii) some of them do not employ public key cryptography such as [18, 20], resulting in that they cannot provide the expected perfect forward secrecy, etc. To overcome these problems, this paper focuses on the design of a secure and efficient two-factor AKE protocol for WSNs in the IoT applications. Our presented two-factor AKE scheme is based on the elliptic curve cryptography (ECC). Specifically, it makes use of a ephemeral secret value to achieve the mutual authentication and session key agreement between the end user and sensor node under the help of the GWN. Unlike the relevant scheme [15], the new AKE protocol does not need to rely on PKI certificates, and it has been analyzed to be able to provide the expected security attributes including forward secrecy, privacy preserving etc., and resist various known attacks. Moreover, the new protocol is efficient and has certain advantages in terms of the computation and communication costs compared with the same type of ECC-based AKE scheme [1].

1.1 Related works

In the past, many AKE protocols, such as [7, 9], have been standardized by authorities worldwide to provide security assurance on the Internet. However, these standardized AKE protocols mainly consider the fundamental two-party communication situations, making them most likely need to be performed more than one time to secure the networks involving at least three parties such as the WSNs. Thus, much more computation and communication overheads will be consumed. Moreover, these standard AKE protocols generally need to depend on the PKI system, which will inevitably involve the use of digital certificates. As we all known, the sensor nodes in the WSNs generally are resource-constrained devices, which may have no enough resources to validate, update or manage various certificates. So, the certificate-based AKE schemes are not very suitable to be deployed in the WSNs. Therefore, many researchers devote to design special secure and efficient AKE protocols for WSNs that do not require the sensor nodes to maintain certificates. Roughly speaking, these presented AKE protocols for WSNs can be divided into two categories, i.e., one type is based on the public key cryptography, e.g., [15–17], etc., while the other type does not depend on PKI certificates, e.g., [22–24], etc.

Although many specific AKE schemes have been presented for securing WSNs in the IoT applications, and some research works, e.g., [12, 14], etc., have investigated in this area, designing a secure and efficient AKE solution is still in progress. Specially, Li et al. [10] designed a temporal-credential-based authentication and session key agreement protocol, while this protocol was pointed out by Kumari et al. [8] to be insecure against the password guessing, stolen verifier and user impersonation attacks, etc., then they presented a temporal-credential-based authentication scheme by using chaotic maps. Porambage et al. [13] presented a two-phase authentication protocol for WSNs in distributed IoT applications, with the merits that allowing the end users and sensor nodes directly to authenticate each other and establish a secure communication link between them. Nevertheless, this scheme was marked by Challa et al. [1] to be unable to provide user anonymity or resist the user impersonation, privileged-insider, replay, denial-of-service and man-in-the-middle attacks. Turkanović et al. [18] also presented an authentication with key agreement protocol for WSNs under the IoT notion, while this scheme was also unfortunately marked to be vulnerable to the off-line password guessing, privileged-insider and user impersonation attacks in Challa et al.'s work [1]. Then, a signature-based authenticated key establishment protocol for IoT application was presented by Challa et al. [1].

Kalra and Sood [6] put forward an AKE protocol for establishing a secure link between the IoT devices and cloud servers using ECC. However, Chang et al. [2] observed that this scheme cannot achieve mutual authentication and session key agreement at all, and they subsequently presented an improved scheme with the aim to eliminate the security flaws in Kalra and Sood's scheme. Later, Wang et al. [19] reviewed Kalra and Sood's scheme [6] and Chang et al.'s scheme [2], in addition to reconfirming the vulnerabilities in Kalra and Sood's scheme, they also identified that Chang et al.'s improved scheme still has a pitfall that an adversary can impersonate an honest server to deceive a remote end user in this system. Then, they also presented a new authentication scheme based on these two schemes and proved their scheme in the random oracle model. Wazid et al. [20] designed an AKE protocol for generic IoT networks, completely based on symmetric cryptography primitives without using asymmetric cryptography primitives. Thus, once the long-term private key stored in GWN is compromised to an adversary $\mathcal{A}$, then $\mathcal{A}$ can extract the previous established session keys, as a result, this scheme cannot provide perfect forward secrecy. In recent, Sadhukhan et al. [15] also proposed an AKE protocol for IoT applications, while it did not use ephemera secret key for each session and user's identity was directly transmitted in the protocol run, making it be unable to provide perfect forward secrecy and user's privacy-preserving.

1.2 Roadmap of the paper

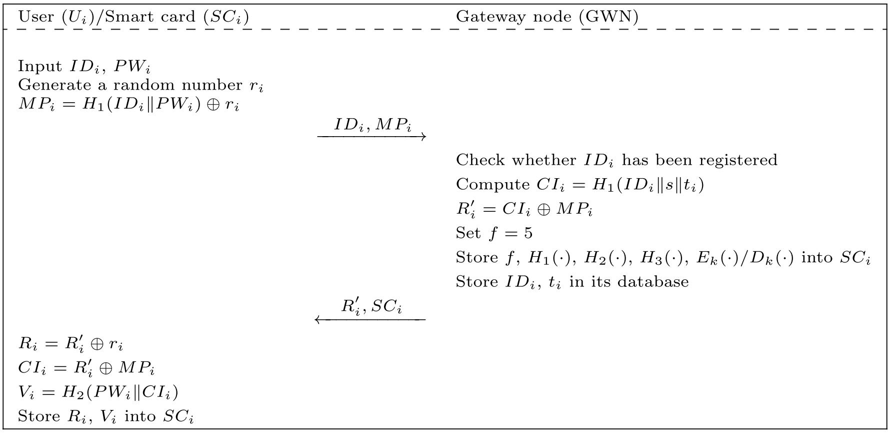
This paper is organized as follows: The designed two-factor AKE protocol for WSNs is detailed in Sect. 2. The security analysis for the new AKE scheme is presented in Sect. 3, and the performance study about it is presented in Sect. 4. Finally, Sect. 5 concludes this paper.

2 The proposed authenticated key exchange protocol

First, the notations throughout this paper are listed in Table 1. The proposed two-factor authenticated key exchange protocol in this paper for WSNs in IoT applications is based on ECC, and consists of the following six phases, where the user registration phase, login and authenticated key exchange phases are also briefly summarized in Figs. 1 and 2, respectively.

Table 1 Notations used in this paper

Notation	Description
GWN	Gateway node
U_i, SN_j	The i-th user and j-th sensor node
$\mathcal{A}$	An adversary
$E_p(a, b)$	An elliptic curve group defined by the equation E: $y^2 = x^3 + ax + b \bmod p$ over the finite filed F_p
G	A generator on $E_p(a, b)$
n	G's big prime order
Z_n^*	The integer set $\{1, 2, \dots, n\}$
P_{pub}, s	GWN's public-private key pair
ID_i, PW_i, SC_i	U_i's identity, password, smart card
$E_k(\cdot)/D_k(\cdot)$	Symmetric encryption/decryption algorithm
$H_1(\cdot), H_2(\cdot), H_3(\cdot)$	Secure one-way hash functions
Δt,f	The preset fault tolerance thresholds
$\|$, $\oplus$	Concatenation and bitwise XOR operations


User (U_i)/Smart card (SC_i)

Gateway node (GWN)

Input ID_i, PW_i
Generate a random number r_i
$MP_i = H_1(ID_i\|PW_i) \oplus r_i$

$\xrightarrow{ID_i, MP_i}$

Check whether ID_i has been registered
Compute $CI_i = H_1(ID_i\|s\|t_i)$
$R_i' = CI_i \oplus MP_i$
Set $f = 5$
Store f, $H_1(\cdot)$, $H_2(\cdot)$, $H_3(\cdot)$, $E_k(\cdot)/D_k(\cdot)$ into SC_i
Store ID_i, t_i in its database

$\xleftarrow{R_i', SC_i}$

$R_i = R_i' \oplus r_i$
$CI_i = R_i' \oplus MP_i$
$V_i = H_2(PW_i\|CI_i)$
Store R_i, V_i into SC_i

Fig. 1 Summary of the user registration phase

2.1 Predeployment phase

In this phase, GWN first chooses the system security parameter ℓ and the secure elliptic curve domain parameters $\{E_p(a, b), G, n\}$, cryptographic hash functions $H_1(\cdot)$: $\{0,1\}^* \rightarrow \{0,1\}^\ell$, $H_2(\cdot)$: $\{0,1\}^* \rightarrow \{0,1\}^\ell$, $H_3(\cdot)$: $\{0,1\}^* \rightarrow \{0,1\}^\ell$, and a secure symmetric encryption/decryption algorithm $E_k(\cdot)/D_k(\cdot)$ (e.g., AES, etc.). Then, GWN randomly chooses a $s \in Z_n^*$ as its master secret key and computes its corresponding public key $P_{pub} = sG$. Next, GWN issues a unique identity SID_j and computes $k_j = H_1(SID_j\|s)$ for each sensor node SN_j to be deployed. Finally, GWN stores $\{SID_j, k_j\}$ into the SN_j prior to its deployment in the network.

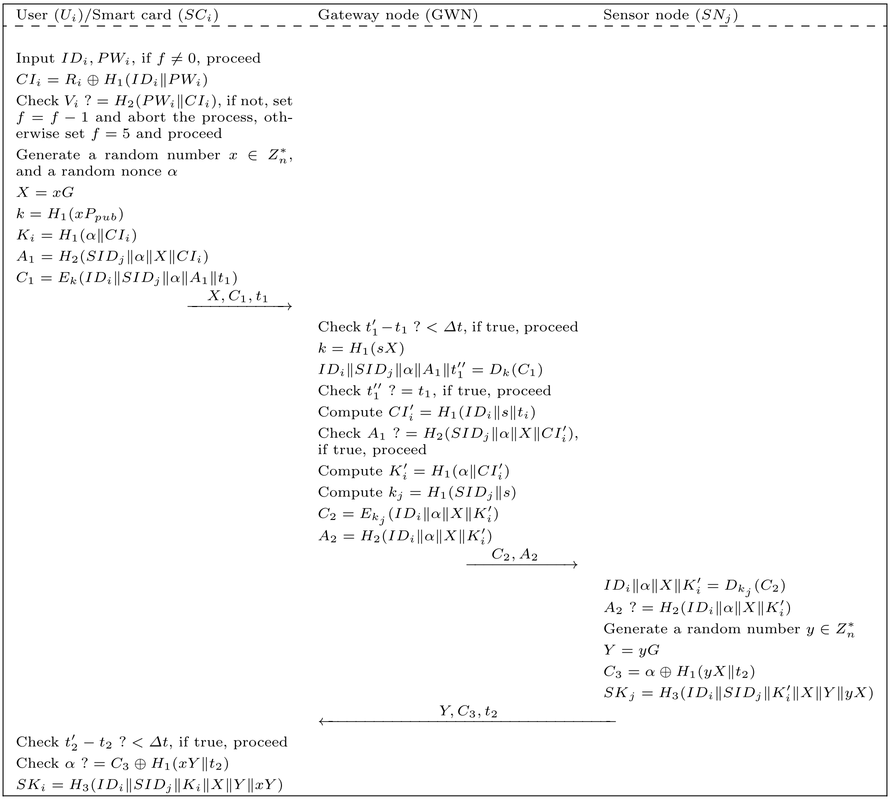

User (U_i)/Smart card (SC_i)
Gateway node (GWN)
Sensor node (SN_j)
Input ID_i, PW_i, if $f \neq 0$, proceed
$CI_i = R_i \oplus H_1(ID_i \| PW_i)$
Check V_i ? $= H_2(PW_i \| CI_i)$, if not, set $f = f - 1$ and abort the process, otherwise set $f = 5$ and proceed
Generate a random number $x \in Z_n^*$, and a random nonce α
$X = xG$
$k = H_1(xP_{pub})$
$K_i = H_1(\alpha \| CI_i)$
$A_1 = H_2(SID_j \| \alpha \| X \| CI_i)$
$C_1 = E_k(ID_i \| SID_j \| \alpha \| A_1 \| t_1)$
$\xrightarrow{X, C_1, t_1}$
Check $t_1' - t_1$? $< \Delta t$, if true, proceed
$k = H_1(sX)$
$ID_i \| SID_j \| \alpha \| A_1 \| t_1'' = D_k(C_1)$
Check t_1'' ? $= t_1$, if true, proceed
Compute $CI_i' = H_1(ID_i \| s \| t_i)$
Check A_1 ? $= H_2(SID_j \| \alpha \| X \| CI_i')$, if true, proceed
Compute $K_i' = H_1(\alpha \| CI_i')$
Compute $k_j = H_1(SID_j \| s)$
$C_2 = E_{k_j}(ID_i \| \alpha \| X \| K_i')$
$A_2 = H_2(ID_i \| \alpha \| X \| K_i')$
$\xrightarrow{C_2, A_2}$
$ID_i \| \alpha \| X \| K_i' = D_{k_j}(C_2)$
A_2 ? $= H_2(ID_i \| \alpha \| X \| K_i')$
Generate a random number $y \in Z_n^*$
$Y = yG$
$C_3 = \alpha \oplus H_1(yX \| t_2)$
$SK_j = H_3(ID_i \| SID_j \| K_i' \| X \| Y \| yX)$
$\xleftarrow{Y, C_3, t_2}$
Check $t_2' - t_2$? $< \Delta t$, if true, proceed
Check α ? $= C_3 \oplus H_1(xY \| t_2)$
$SK_i = H_3(ID_i \| SID_j \| K_i \| X \| Y \| xY)$

Fig. 2 Summary of the login, and authenticated key exchange phases

2.2 User registration phase

To access information from SN_j, U_i should first do the steps below to register on the gateway node GWN.

(1) U_i inputs his/her unique identity ID_i, password PW_i and generates a random number r_i to compute $MP_i = H_1(ID_i \| PW_i) \oplus r_i$, then sends $\{ID_i, MP_i\}$ to the GWN for registration via a secure channel.

(2) On receiving $\{ID_i, MP_i\}$ at the time t_i, GWN checks whether ID_i already exists in its database. If true, GWN reminds U_i to choose another identity, otherwise, GWN computes $CI_i = H_1(ID_i \| s \| t_i)$, $R_i' = CI_i \oplus MP_i$ and sets $f = 5$ for U_i. Then, GWN stores $\{ID_i, t_i\}$ in its database, and stores $\{f, H_1(\cdot), H_2(\cdot), H_3(\cdot), E_k(\cdot)/D_k(\cdot)\}$ into a smart card SC_i. Finally, GWN returns R_i', SC_i back to the U_i securely.

(3) On receiving R_i' and SC_i, U_i computes $R_i = R_i' \oplus r_i$, $CI_i = R_i' \oplus MP_i$, $V_i = H_2(PW_i \| CI_i)$, and stores R_i, V_i into SC_i.

2.3 Login phase

U_i can easily initiate a login request to access the intended SN_j by the steps below.

(1) U_i inputs his/her identity ID_i, password PW_i, if $f \neq 0$, then SC_i extracts $CI_i = R_i \oplus H_1(ID_i \| PW_i)$ and checks whether $V_i = H_2(PW_i \| CI_i)$ holds or not. If not, SC_i sets $f = f - 1$ and aborts the phase, otherwise, sets $f = 5$ and proceeds to do the next step.

(2) SC_i generates a random numbers $x \in Z_n^*$ and a random nonce α, then computes $X = xG$, $k = H_1(xP_{pub})$, $K_i = H_1(\alpha \| CI_i)$ and $A_1 = H_2(SID_j \| \alpha \| X \| CI_i)$, $C_1 = E_k(ID_i \| SID_j \| \alpha \| A_1 \| t_1)$, where t_1 is its current timestamp. Finally, SC_i sends the login request $\{X, C_1, t_1\}$ to GWN.

2.4 Authenticated key exchange phase

On receiving the login request $\{X, C_1, t_1\}$ at the time t_1', GWN performs the steps below to help the end user and sensor node establish a secure communication link.

(1) GWN first validates the freshness of t_1 by checking whether $t_1' - t_1 < \Delta t$ holds or not. If not, GWN terminates the session, otherwise, GWN proceeds to compute $k = H_1(sX)$, $ID_i \| SID_j \| \alpha \| A_1 \| t_1'' = D_k(C_1)$ and checks whether $t_1'' = t_1$ holds or not. If not, GWN terminates the session, otherwise, GWN proceeds to compute $CI_i' = H_1(ID_i \| s \| t_i)$ by using its private key s and the retrieved t_i from its database and checks whether $A_1 = H_2(SID_j \| \alpha \| X \| CI_i')$ holds or not. If not, GWN terminates the session, otherwise, proceeds to compute $K_i' = H_1(\alpha \| CI_i')$, $k_j = H_1(SID_j \| s)$, $C_2 = E_{k_j}(ID_i \| \alpha \| X \| K_i')$, and $A_2 = H_2(ID_i \| \alpha \| X \| K_i')$. Finally, GWN sends $\{C_2, A_2\}$ to SN_j.

(2) On receiving $\{C_2, A_2\}$, the sensor node SN_j computes $ID_i \| \alpha \| X \| K_i' = D_{k_j}(C_2)$ using its secret key k_j and checks whether $A_2 = H_2(ID_i \| \alpha \| X \| K_i')$ holds or not. If not, SN_j aborts this session, otherwise generates a random nonce $y \in Z_n^*$ to compute $Y = yG$ and uses its current timestamp t_2 to compute $C_3 = \alpha \oplus H_1(yX \| t_2)$, the session key $SK_j = H_3(ID_i \| SID_j \| K_i' \| X \| Y \| yX)$. Then, SN_j sends $\{Y, C_3, t_2\}$ to U_i.

(3) On receiving $\{Y, C_3, t_2\}$ at the time t_2', U_i validates the freshness of t_2 by checking whether $t_2' - t_2 < \Delta t$ holds or not. If not, U_i terminates the session, otherwise, proceeds to check whether $\alpha = C_3 \oplus H_1(xY \| t_2)$ holds or not. If not, U_i aborts the session, otherwise computes the session key $SK_i = H_3(ID_i \| SID_j \| K_i \| X \| Y \| xY)$.

2.5 Password update phase

In the proposed AKE protocol, a legitimate user U_i can update his/her password at any time locally by doing as the steps below without connecting to the network.

(1) U_i inputs his/her identity ID_i, password PW_i, if $f = 0$, SC_i aborts this procedure; if $f \neq 0$, then SC_i extracts $CI_i = R_i \oplus H_1(ID_i \| PW_i)$ and checks whether

$V_i = H_2(PW_i \| CI_i)$ holds or not. If not, SC_i sets $f = f - 1$ and aborts the process, otherwise, asks U_i to enter a new password PW_i^*.

(2) After inputting the new password PW_i^*, SC_i computes $R_i^* = CI_i \oplus H_1(ID_i \| PW_i^*)$ and $V_i^* = H_2(PW_i^* \| CI_i)$. Then, SC_i sets $f = 5$ and replaces R_i and V_i with R_i^* and V_i^*, respectively.

2.6 Dynamic sensor node deployment phase

Deploying a new sensor node, say SN_j^*, in the existing network to expand the scope of service may be an inevitable demand in practice. In the new scheme, GWN can realize this demand by doing as the following steps.

(1) GWN assigns a new unique identity SID_j^* for SN_j^* and computes $k_j^* = H_1(SID_j^* \| s)$. Then, GWN stores k_j^* into SN_j^* prior to its deployment.

(2) GWN displays the new deployed sensor node SN_j^* to users so that they can access some information from SN_j^*, if needed.

Then, when a legitimate user wants to access information from the new node SN_j^*, he/she just needs to perform the authenticated key exchange phase with SN_j^* to accomplish mutual authentication and session key agreement, and then gain information from the node SN_j^* securely.

3 Security analysis

This section gives a heuristic security analysis to the presented two-factor AKE scheme for WSNs, which demonstrates that the new scheme supports various security attributes and is secure against various known attacks.

3.1 The new scheme supports mutual authentication and session key agreement

In the new scheme, the gateway node GWN can first authenticate the legitimate of the user U_i by checking whether $A_1 = H_2(SID_j \| \alpha \| X \| CI_i')$ holds or not, since only the legal user U_i can derive the corresponding valid credential CI_i using his/her PW_i and SC_i, and use it to compute the valid A_1. Then, the sensor node can authenticate the GWN by checking whether $A_2 = H_2(ID_i \| \alpha \| X \| K_i')$ holds or not, since only the legal GWN can derive the same secret k_j shared between them and use it to compute the valid ciphertext C_2 and the authentication tag A_2. Moreover, the ephemeral secret nonce α generated by U_i for each session is encrypted in the ciphertext C_2, only the legal sensor node can decrypt C_2 using its private key k_j to get it, thus the sensor node can use it to authenticate itself to the user, i.e., U_i can authenticate the sensor node by checking whether $\alpha = C_3 \oplus H_1(xY \| t_2)$ holds or not. Thus, when the legitimate of the user and sensor node are authenticated successfully, it is obvious that they can derive the same session key $SK = H_3(ID_i \| SID_j \| K_i \| X \| Y \| xyG)$.

3.2 The new scheme supports perfect forward secrecy

In the new scheme, the secret value xyG is required to compute the session key, i.e., $SK = H_3(ID_i \| SID_j \| K_i \| X \| Y \| xyG)$, which involves the specific random numbers x and y different in each session and contributed by the user U_i and the corresponding sensor node, respectively. Therefore, even if the long-term secret keys held by the user, GWN and sensor node are all compromised, the previous established session keys remain secure if the corresponding ephemeral secrets x and y are not revealed. Meanwhile, the well-known elliptic curve Diffie–Hellman (ECDH) security assumption ensures that an adversary $\mathcal{A}$ cannot derive the secret xyG even if it may intercept the ephemeral public keys $X = xG$ and $Y = yG$. Therefore, the perfect forward secrecy is supported in the new scheme.

3.3 The new scheme achieves user's privacy-preserving

In the new scheme, the user U_i's identity ID_i is encrypted in the ciphertext C_1 and then transmitted to the GWN. Then, the GWN re-encrypts U_i's identity ID_i in the ciphertext C_2 and then transmits it to the intended sensor node. So, for any adversary $\mathcal{A}$, it cannot decrypt the ciphertexts to reveal U_i's identity ID_i without having the corresponding secret keys. Moreover, these ciphertexts are dynamic due to that the ephemeral secret nonce α for each session is also encrypted in these ciphertexts. So, the user U_i's anonymity is always kept and the user cannot be traced. Therefore, user's privacy-preserving is achieved in the new scheme.

3.4 The new scheme can resist the off-line password guessing attack

Assume a legal user U_i's smart card SC_i is obtained by an adversary $\mathcal{A}$ by some ways, then $\mathcal{A}$ may try to make use of it to perform the off-line password guessing attack by trying to enumerate the password dictionary, while it should be noted that a fault-tolerant value f is set in the SC_i, and every wrong attempt will reduce its value by one and when $f = 0$ then SC_i will lock immediately, thus $\mathcal{A}$ cannot try other passwords. Therefore, the off-line password guessing attack is resisted in the new scheme.

3.5 The new scheme can resist the impersonation attack

Obviously, to impersonate a legal user U_i to pass the verification test performed by the GWN, an adversary $\mathcal{A}$ must send a valid login message $\{X, C_1, t_1\}$ to the GWN. However, the two-factor authentication security assumption makes it be impossible for $\mathcal{A}$ to get the secret CI_i to compute the valid login request messages to deceive GWN successfully. Moreover, without the secret key k_j, $\mathcal{A}$ cannot compute the valid $\{C_2, A_2\}$ to deceive the sensor node, so $\mathcal{A}$ cannot impersonate the

GWN to the sensor node since extracting k_j needs the private key s of GWN or compromising the corresponding sensor node which are assumed impossible for $\mathcal{A}$. Meanwhile, it should be noted that without the corresponding secret key k_j, $\mathcal{A}$ cannot decrypt the ciphertext C_2 to obtain the secret nonce α, then it cannot compute the valid C_3 to pass the user's verification test. Therefore, the impersonation attack is resisted in the new scheme.

3.6 The new scheme can resist the replay attack

In the new scheme, it can be obviously seen that the timestamp and random nonce are embedded in the mutual authentication and session key agreement phase. If an adversary $\mathcal{A}$ replays a previously intercepted messages $\{X, C_1, t_1\}$ of a user to the GWN, then the inequality $t_1' - t_1 > \Delta t$ will most likely hold and thereby the session will be immediately aborted. Moreover, even if the timestamp t_1 is modified by $\mathcal{A}$ to satisfy $t_1' - t_1 < \Delta t$, GWN will detect whether the received timestamp has been modified or not when it decrypts the ciphertext C_1 since the original timestamp is also encrypted in the ciphertext C_1. Similarly, when the adversary replays the messages $\{C_2, A_2\}$ or $\{Y, C_3, t_2\}$ to the corresponding receiver, the user will detect the replay attack by checking whether $\alpha = C_3 \oplus H_1(xY \| t_2)$ holds or not. Therefore, the replay attack is resisted in the new scheme.

3.7 The new scheme can resist the man-in-the-middle attack

The man-in-the-middle attack implies that an adversary $\mathcal{A}$ can independently communicate with the legal user U_i, the GWN and the sensor node, and transmit messages between them, making them mistakenly believe that they are communicating directly with the correct peer parties. In other words, it requires $\mathcal{A}$ to be able to impersonate one party to its peer party successfully. However, according to the Sects. 3.5 and 3.6, $\mathcal{A}$ can mount neither the impersonation attack nor the replay attack, thus the man-in-the-middle attack cannot be executed successfully. Therefore, the new scheme can resist the man-in-the-middle attack.

3.8 The new scheme can resist the privileged insider attack

In the registration phase of the new scheme, U_i submits his/her ID_i and the computed $MP_i = H_1(ID_i \| PW_i) \oplus r_i$ to the gateway node GWN via a secure channel. It is obvious that GWN cannot know U_i's password PW_i or the secret $H_1(ID_i \| PW_i)$ by MP_i. Thus, even if an adversary $\mathcal{A}$ is a privileged insider, it cannot obtain user's sensitive secret information so as to perform other attacks. Therefore, the privileged insider attack is resisted in the new scheme.

Table 2 Security feathers comparison

Security features	Scheme				
	[1]	[15]	[18]	[20]	New
Mutual authentication and session key agreement	√	√	√	√	√
Perfect forward secrecy	√	×	×	×	√
Privacy-preserving	√	×	×	√	√
Off-line password guessing attack resistance	√	√	×	√	√
Replay attack resistance	√	√	√	√	√
Impersonation attack resistance	√	√	√	√	√
Man-in-the-middle attack resistance	√	√	√	√	√
Privileged insider attack resistance	√	√	×	√	√

4 Performance study

This section compares the performance of the designed two-factor AKE scheme with some recent related works, i.e. Challa et al. [1], Sadhukhan et al. [15], Turkanović et al. [18] and Wazid et al. [20], in the aspects of security features, computation and communication costs.

The security features of these related existing schemes and the presented AKE protocol in this work are summarized in Table 2, from which it can be easily observed that our new scheme and Challa et al.'s scheme [1] can resist various well-known attacks, while Sadhukhan et al.'s scheme [15], Turkanović et al.'s scheme [18] and Wazid et al.'s scheme [20] are vulnerable to some attacks, respectively. Although Sadhukhan et al.'s scheme [15] employed public key cryptography as its underlying security technology, it did not involve ephemeral secret key for each new session, resulting in it unable to provide perfect forward secrecy. Moreover, Sadhukhan et al.'s scheme cannot preserve user's privacy since user's identity is directly transmitted in this scheme. Turkanović et al.'s scheme [18] and Wazid et al.'s scheme [20] are just based on symmetric cryptography primitives. As a result, their schemes at least cannot provide perfect forward secrecy.

To denote the time consumption of different cryptographic operations succinctly, the notations below are used:

Table 3 Computation Costs Comparison

Scheme	U_i	GWN	SN_j
[1]	$5T_{pm} + 1T_{fe} + 8T_h \approx 382.45$ ms	$5T_{pm} + 4T_h \approx 317.375$ ms	$4T_{pm} + 3T_h \approx 253.8$ ms
[15]	$1T_{pm} + 2T_{ed} + 1T_h \approx 80.975$ ms	$4T_{ed} + 2T_h \approx 35.8$ ms	$1T_{pm} + 2T_{ed} + 1T_h \approx 80.975$ ms
[18]	$7T_h \approx 3.5$ ms	$5T_h \approx 2.5$ ms	$7T_h \approx 3.5$ ms
[20]	$1T_{fe} + 2T_{ed} + 13T_h \approx 86.975$ ms	$4T_{ed} + 5T_h \approx 37.3$ ms	$2T_{ed} + 4T_h \approx 19.4$ ms
New	$3T_{pm} + 1T_{ed} + 7T_h \approx 201.425$ ms	$1T_{pm} + 2T_{ed} + 6T_h \approx 83.475$ ms	$2T_{pm} + 1T_{ed} + 3T_h \approx 136.35$ ms

- T_{pm} denotes the time for executing an elliptic curve scalar point multiplication operation;
- T_{ed} denotes the time for executing an encryption/decryption operation;
- T_{fe} denotes the time for executing a fuzzy extractor operation (used in [1, 20]);
- T_h denotes the time for executing a cryptographic hash operation.

Then, the computation costs for the login and authenticated key exchange phases of these related existing schemes and our new AKE protocol are summarized in Table 3, where the experiment results used in [20], i.e., $T_{pm} \approx 63.075\,\text{ms}$, $T_{ed} \approx 8.7\,\text{ms}$, $T_{fe} \approx 63.075\,\text{ms}$ (under the assumption that $T_{fe} \approx T_{pm}$) and $T_h \approx 0.5\,\text{ms}$, are referenced here for intuitive evaluation. From Table 3, it can be easily observed that the computation costs required for each U_i, GWN and SN_j are $3T_{pm} + 1T_{ed} + 7T_h \approx 201.425\,\text{ms}$, $1T_{pm} + 2T_{ed} + 6T_h \approx 83.475\,\text{ms}$ and $2T_{pm} + 1T_{ed} + 3T_h \approx 136.35\,\text{ms}$, respectively, in our new scheme, which consume much less time than the same type of ECC-based AKE scheme [1]. The schemes in [18] and [20] consume less time than our new scheme and the scheme [1], mainly because of that they do not use public key cryptography to ensure security, but which also results in the infeasibility of them to provide forward secrecy. Sadhukhan et al.'s scheme [15] consumes less time than our new scheme since it used pre-computed ECDH-based symmetric keys shared between the corresponding parties to reduce some computations, but this practice makes this scheme unable to provide perfect forward secrecy. In addition, these keys need to be updated in a regular time interval.

In terms of the communication costs, assume the bit lengths of the identity and timestamp are both 64 bits, the block length of the used symmetric cryptographic algorithm (e.g., AES) is 128 bits, the bit lengths of the hash digest and random nonce are both 160 bits, and the bit length of an elliptic curve point is 320 bits, then the messages exchanged between the U_i, GWN and SN_j in our new scheme are 896 bits, 928 bits and 544 bits, respectively, and thus the total communication costs are 2368 bits. Moreover, our new scheme just requires 3 communication flows to achieve mutual authentication and session key agreement between the end user and sensor node. Similarly, the total communication costs and flows of our new scheme and the other related schemes are also summarized in Table 4, from which it can be easily observed that our new scheme still has some advantages in terms of communication overhead, especially compared with the same type of ECC-based schemes [1] and [15].

Table 4 Communication costs comparison

Scheme	Communication rounds	Total costs
[1]	3	2496 bits
[15]	4	3712 bits
[18]	4	2944 bits
[20]	4	2784 bits
Our	3	2368 bits

5 Conclusion

In this paper, a new two-factor authenticated key exchange protocol for wireless sensor networks in IoT applications is presented with the aim to contribute some to the IoT security. The designed protocol is based on ECC and allows the end user to exchange information with the sensor node directly after a secure link is established. The heuristic security analysis in this paper confirms that the proposed protocol has perfect security properties and can withstand various well-known attacks. Moreover, the performance study of computation and communication costs demonstrates that the proposed protocol has certain advantages compared with the same type of ECC-based schemes. So, the presented protocol may be a more suitable one for securing the WSNs in IoT environments. In addition, how to make use of the pairing-based public key cryptography to design secure and efficient AKE protocols for IoT applications is one of our future works related with this paper.

Acknowledgements The authors would like to sincerely thank the anonymous reviewers for their valuable comments, and the editor for the patience and hard work for this paper. This work was supported in part by the Natural Science Basic Research Program of Shaanxi Province of China under Grant 2021JQ-123, and in part by the Fundamental Research Funds for the Central Universities under Grant 31020200QD011.

References

1. Challa S, Wazid M, Das AK, Kumar N, Reddy AG, Yoon EJ, Yoo KY (2017) Secure signature-based authenticated key establishment scheme for future IoT applications. IEEE Access 5(99):3028–3043
2. Chang CC, Wu HL, Sun CY (2016) Notes on secure authentication scheme for IoT and cloud servers. Pervasive Mob Comput 38
3. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of things (IoT): a vision, architectural elements, and future directions. Futur Gener Comput Syst 29(7):1645–1660
4. Gutmann P (2002) PKI: it's not dead, just resting. Computer 35(8):41–49
5. Islam SMR, Kwak D, Kabir MH, Hossain M, Kwak KS (2017) The internet of things for health care: a comprehensive survey. IEEE Access 3:678–708
6. Kalra S, Sood SK (2015) Secure authentication scheme for IoT and cloud servers. Pervasive Mob Comput 24:210–223
7. Krawczyk H (2005) HMQV: a high-performance secure Diffie-Hellman protocol. Crypto 3621:546–566
8. Kumari S, Wu F, Wu F, Das AK, Arshad H, Khan MK (2016) A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps. Futur Gener Comput Syst 63(C):56–75
9. Law L, Menezes A, Qu M, Solinas J, Vanstone S (2003) An efficient protocol for authenticated key agreement. Des Codes Cryptogr 28(2):119–134
10. Li CT, Weng CY, Lee CC (2013) An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks. Sensors 13(8):9589–9603
11. Miorandi D, Sicari S, Pellegrini FD, Chlamtac I (2012) Internet of things: vision, applications and research challenges. Ad Hoc Netw 10(7):1497–1516
12. Nguyen KT, Laurent M, Oualha N (2015) Survey on secure communication protocols for the internet of things. Ad Hoc Netw 32(2):17–31
13. Porambage P, Schmitt C, Kumar P, Gurtov A, Ylianttila M (2014) Two-phase authentication protocol for wireless sensor networks in distributed IoT applications. In: 2014 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, pp 2728–2733

14. Roman R, Alcaraz C, Lopez J, Sklavos N (2011) Key management systems for sensor networks in the context of the internet of things. Comput Electr Eng 37(2):147–159
15. Sadhukhan D, Ray S, Biswas GP, Khan MK, Dasgupta M (2020) A lightweight remote user authentication scheme for IoT communication using elliptic curve cryptography. J Supercomput 1–38 (2020)
16. Simplicio MA Jr, Silva MV, Alves RC, Shibata TK (2017) Lightweight and escrow-less authenticated key agreement for the internet of things. Comput Commun 98:43–51
17. Ting P, Tsai J, Wu T (2018) Signcryption method suitable for low-power IoT devices in a wireless sensor network. IEEE Syst J 12(3):2385–2394
18. Turkanović M, Brumen B, Hölbl M (2014) A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion. Ad Hoc Netw 20(2):96–112
19. Wang KH, Chen CM, Fang W, Wu TY (2017) A secure authentication scheme for internet of things. Pervasive Mob Comput 42
20. Wazid M, Das AK, Odelu V, Kumar N, Conti M, Jo M (2018) Design of secure user authenticated key management protocol for generic IoT networks. IEEE Internet Things J 5(1):269–282
21. Wollschlaeger M, Sauter T, Jasperneite J (2017) The future of industrial communication: automation networks in the era of the internet of things and industry 4.0. IEEE Ind Electron Mag 11(1):17–27
22. Xu L, Wu F (2019) A lightweight authentication scheme for multi-gateway wireless sensor networks under IoT conception. Arab J Sci Eng 44(4):3977–3993
23. Xue K, Ma C, Hong P, Ding R (2013) A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. J Netw Comput Appl 36(1):316–323
24. Yang SK, Shiue YM, Su ZY, Liu IH, Liu CG (2020) An authentication information exchange scheme in WSN for IoT applications. IEEE Access 8:9728–9738
25. Zanella A, Bui N, Castellani A, Vangelista L, Zorzi M (2014) Internet of things for smart cities. IEEE Internet Things J 1(1):22–32

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.