Computer Networks 73 (2014) 41–57

Contents lists available at ScienceDirect

Computer Networks

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

On the anonymity of two-factor authentication schemes for wireless sensor networks: Attacks, principle and solutions

Ding Wang [a,b,*], Ping Wang [b,c]

[a] School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
[b] National Engineering Research Center for Software Engineering, Beijing 100871, China
[c] School of Software and Microelectronics, Peking University, Beijing 100260, China

ARTICLE INFO

Article history:
Received 2 April 2014
Received in revised form 27 June 2014
Accepted 27 July 2014
Available online 11 August 2014

Keywords:
Two-factor authentication
Wireless sensor networks
User anonymity
Smart card
Non-tamper resistant

ABSTRACT

Anonymity is among the important properties of two-factor authentication schemes for wireless sensor networks (WSNs) to preserve user privacy. Though impressive efforts have been devoted to designing schemes with user anonymity by only using lightweight symmetric-key primitives such as hash functions and block ciphers, to the best of our knowledge none has succeeded so far. In this work, we take an initial step to shed light on the rationale underlying this prominent issue. Firstly, we scrutinize two previously-thought sound schemes, namely Fan et al.'s scheme and Xue et al.'s scheme, and demonstrate the major challenges in designing a scheme with user anonymity.

Secondly, using these two foremost schemes as case studies and on the basis of the work of Halevi–Krawczyk (1999) [44] and Impagliazzo–Rudich (1989) [43], we put forward a general principle: Public-key techniques are intrinsically indispensable to construct a two-factor authentication scheme that can support user anonymity. Furthermore, we discuss the practical solutions to realize user anonymity. Remarkably, our principle can be applied to two-factor schemes for universal environments besides WSNs, such as the Internet, global mobility networks and mobile clouds. We believe that our work contributes to a better understanding of the inherent complexity in achieving user privacy, and will establish a groundwork for developing more secure and efficient privacy-preserving two-factor authentication schemes.

1. Introduction

With the rapid development of micro-electromechanical systems and wireless network technologies, wireless sensor networks (WSNs) have attracted increasing attention due to its wide range of applications from battlefield surveillance to civilian applications, e.g., environmental monitoring, real-time traffic control, industrial process control and home automation. As is well known, most large-scale WSNs [1–3] follow a tiered architecture due to its superiority in increasing the network capacity and scalability, accommodating the node mobility, reducing the management complexity and prolonging the network lifetime. Thus, in this work we mainly focus on the tiered WSNs as well. In many critical applications, external users are generally interested in accessing real-time information from sensor nodes, yet if the data queries are issued by the base station, efficiency, scalability and security may not be ensured over the long communication path between the base station and the sensor nodes [4,5].

* Corresponding author at: Room 560, 2#, Changchunxinyuan, University, No. 5 Yiheyuan Road, Haidian District, Beijing 100871, China. Tel.: +86 185 1134 5776; fax: +86 010 6276 5808.
E-mail addresses: wangdingg@mail.nankai.edu.cn (D. Wang), pwang@pku.edu.cn (P. Wang).

http://dx.doi.org/10.1016/j.comnet.2014.07.010
1389-1286/

To enable external users to access the real-time data directly from the desired sensor nodes without involving the gateway node (or base station) as demanded, it is of great concern that such critical data is well protected from eavesdropping, malicious modification, unauthorized access, and so on. Accordingly, user authentication constitutes an essential security mechanism for the user to be first authenticated by the sensor nodes before being granted the right to access data. Owing to its simplicity, portability, efficiency and high level of security, smart-card-based password authentication (or the so-called two-factor authentication [6]), as depicted in Fig. 1, has become one of the most promising authentication mechanisms for real-time data access in WSNs.

The past twenty years of research on two-factor authentication has proved that, it is incredibly difficult to get a general-purpose two-factor scheme right [7–9]. The design of a secure and efficient scheme for WSNs can only be harder. Crucially, the designers are confronted with a paradoxical challenge—"providing lightweight cryptographic algorithms for strong authentication, privacy and other cryptographic services on a speck of dust" [10]. On the one hand, sensor nodes and smart cards are small devices with low computation capability, limited memory capacity and scarce energy resources, it is more desirable to only employ symmetric-key techniques (e.g., hash functions, symmetric encryptions and XOR operations) rather than to use comparatively expensive asymmetric cryptographic operations (e.g., modular exponentiation and Pairing).

On the other hand, WSNs are generally deployed in unattended environments and often perform extremely sensitive tasks (e.g., health-care and battlefield surveillance) and thus, in addition to the traditional security threats, they exhibit a larger attack surface and are prone to more serious (even life-threatening) attacks. Consequently, an admired two-factor authentication scheme for WSNs should be able to guard against various known attacks including these general attacks such as impersonation, replay and offline password guessing, as well as some special attacks in WSNs environments like gateway bypassing and node capture [11]. Besides security, user privacy is also of particular interest. For example, some current projects including GEOSS [12] and NOPP [13] are developing large-scale WSNs to adaptively monitor the earth–ocean–atmosphere system. The sensed data may be of interest to various types of users ranging from individual users to universities, government research centers, and business companies (e.g., GEOSS [12] involves 61 countries, NOPP [13] involves the DARPA, the Department of Homeland Security among others). The activities of these users may be of great sensitiveness to the outsiders and even the users themselves cannot fully trust each other due to diversified interests. Consequently, there is an urgent need for protecting user's data access privacy, e.g., when she accessed the sensor data, which data types she was interested in, or from which nodes she obtained the data, since the leakage of such information could be exploited against her interest. Generally, there is a growing requirement for protecting user privacy information (e.g., preferences, login history, location, physical condition, personal data [14,15]) from being leaked and abused, which outlines the needs for designing schemes that can attain user anonymity.

It is worth mentioning that, in the context of user authentication, user anonymity is defined against the public rather than the server, because it is necessary for the latter to be aware of the real identity of each user in order to detect, record and remove the malicious users. Moreover, in many cases the server needs to learn the user identity for accounting, auditing, and/or billing purposes [16]. It also should be noted that, instead of a unique "user anonymity" property, different application scenarios may implement quite varied notions of what it means to be user anonymity [17,18], such as user identity protection, user un-traceability, anonymous user linkability, k-anonymity and blender anonymity. Interested readers are referred to [19] for more details. As for user authentication, this notion basically means user identity-protection, which ensures

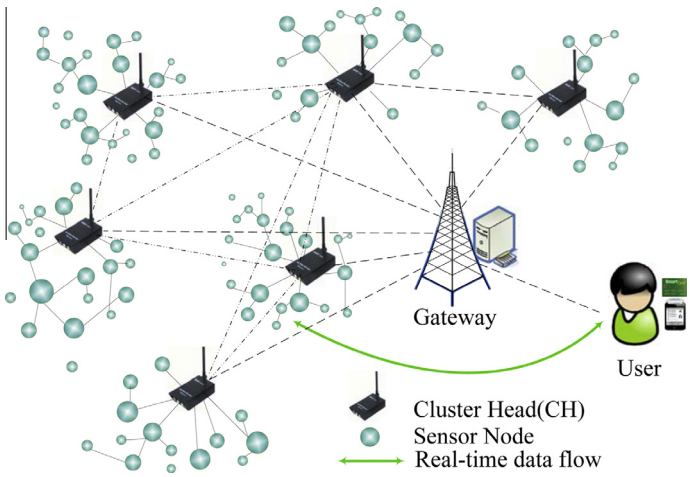

Fig. 1. Direct real-time application data access in WSNs.

that the adversary cannot figure out the target user's identity from the protocol transcripts.

Comparatively, a more satisfactory property of user anonymity is user un-traceability, which guarantees that the adversary can neither determine who the user is nor tell apart whether two conversations originate from the same (unknown) user [20,21]. That is, a scheme achieving this advanced property can prevent the adversary from linking multiple instances of communication generated by the same user and from tracing a user's current location, moving history, etc. Consequently, most schemes (e.g., [11,20,22–28]) that attempt to preserve user privacy aim at fulfilling this stronger notion of user anonymity. Throughout this paper, unless otherwise specified, by "user anonymity" we will always mean "user un-traceability".

1.1. Related work

In 2009, Das [11] suggested the first smart-card-based password authentication scheme to provide mutual authentication among the external user, gateway node and the sensor node. Since this scheme only involves symmetric-key operations, it facilitates authorized users to access WSNs via low-powered mobile devices (e.g., PDAs, Smart phones and Laptops). However, shortly after this two-factor authentication scheme was presented, it is found prone to various attacks, such as offline password guessing attack and sensor node compromise attack by Nyang–Lee [29], gateway node by-passing attack by Khan–Alghath [24], impersonation attack and insider attack by He et al. [23] and Chen and Shih [30]. To eliminate these security drawbacks in Das's scheme, a number of improvements [23,24,29–31] were further proposed. To the designers' disappointment, these five enhancements still cannot attain the claimed security and have been invariably shown flawed [32–35].

In 2011, Fan et al. [36] observed that previous two-factor authentication schemes for real-time data access in WSNs have various defects overlooked, and proposed a new privacy-preserving scheme, which involves only lightweight operations, such as one-way hash function and exclusive-OR operations. Hence, it is well-suited to the resource-limited sensor networks and exhibits great potential for practical use. The authors claimed that their scheme is free from various related cryptographic attacks and can preserve user anonymity. Although their scheme has many merits over existing schemes and even has been equipped with a formal security proof, as will be showed in Section 2, it actually cannot support user anonymity and is vulnerable to several practical attacks.

At the meantime with Kumar et al. [25] also noticed that previous schemes either are subject to security defects or short of essential features like mutual authentication and user anonymity. Therefore, they made an attempt to develop a privacy-preserving two-factor authentication framework for WSNs that can withstand all known attacks and support various features. Shortly after this framework was proposed, Jiang et al. [26] pointed out that it cannot resist offline password guessing attack and fails to provide user un-traceability, and to overcome these two drawbacks, an enhancement was suggested. However, it is not difficult to see that Jiang et al.'s scheme [26] is vulnerable to de-synchronization attack – a dishonest (comprised) sensor node can render the victim user's smart cards completely un-usable by simply altering the last message flow without detection.

In 2012, Das et al. [33] introduced a novel scheme that can support dynamic nodes addition after the initial deployment of nodes in the existing sensor network. To keep efficiency and suitability for being implemented in resource-constrained sensor nodes, this scheme also only involves lightweight operations, such as hash functions and symmetric-key encryption. Yet it was subsequently found prone to a serious design flaw by Turkanovic and Holbl [37], who further proposed an improved version. Unfortunately, Li et al. [38] recently reported that both schemes in [33,37] are susceptible to a flaw that may easily leak the server's long-term secret key.

In 2013, Xue et al. [27] described a lightweight temporal-credential-based mutual authentication and key agreement scheme. As with previous schemes [11,23,25,26,36], this protocol[1] also attempts to achieve the property of user anonymity by only involving hash and XOR operations. Xue et al.'s scheme can provide relatively more admired features and comparatively higher security assurance such as friendliness of password update and *GWN* bypassing attack resilience over existing schemes, yet non-withstanding their long list of security arguments, we will demonstrate that it still cannot achieve its essential goal of user anonymity.

In general, the above unending failure in preserving user anonymity may be largely attributed to (and intrinsically rooted in) the lack of impossibility results in cryptography. Existing results are mainly related to cryptographic primitives and certain proof methods [39]. For example, some impossibility results on using black-box methods for constructing one primitive from another one were reported in [40–42]. At the protocol level, results are more scarce. In 1989, Impagliazzo and Rudich [43] reported the well-known result that key-exchange protocols are unlikely to be obtained by using only symmetric-key techniques. Based on this work, in 1999 Halevi and Krawczyk [44] proved that only symmetric-key techniques are inadequate for password protocols that resist off-line dictionary attacks. In 2000, Park et al. [45] observed that public-key techniques are highly unavoidable for providing forward secrecy in key establishment protocols. In 2006, Nguyen [46] investigated the relationship between password protocols and other cryptographic primitives, and observed that password-authenticated key exchange and public-key encryption are incomparable under black-box reductions. In 2011, González Muñiz and Laud [39] confirmed that only symmetric-key techniques are insufficient to construct message recognition protocols with perenniality.

Among the aforementioned theoretical studies, Halevi–Krawczyk's work [44] may be the closest to what we will discuss in the current paper, however, it only deals with one-factor (i.e., the password factor) and mainly from a

[1] Note that "protocol" and "scheme" will be used interchangeably throughout this paper.

security perspective. In 2009, Zeng et al. [47] pointed out that three previous two-factor schemes for wireless mobile networks [48–50] are unlikely to provide anonymity if an attacker, who is also registered as a legitimate user, can extract secret data from her own smart card.[2] As far as we know, it is the only work that focuses on user anonymity in two-factor authentication. Nonetheless, it blames the failure of these schemes merely on "structural mistakes", leaving untouched the foundational rationale underlying the repeated anonymity failure. To the best of our knowledge, little attention has been given to *the inherent complexity* of designing a two-factor authentication scheme with user anonymity.

1.2. Motivations

There have recently been a number of works (e.g., [24–27,29,30,35,36,51,52]) that endeavor to construct practical two-factor authentication schemes for WSNs, yet few has succeeded in achieving the "precious" property of user anonymity so far. One common feature of these proposals is that, they attempt to preserve user anonymity by only using symmetric-key cryptographic primitives and relying on the non-tamper-resistance assumption about the smart cards. While their incentive to only employ symmetric-key techniques is quite obvious, the non-tamper-resistance assumption about smart cards deserves special attention. Recent research results have demonstrated that the security parameters stored in common commercial smart cards could be revealed or partially extracted by the state-of-the-art side-channel attacks (e.g., differential power analysis) [53–55], software attacks (launched on software-supported card, e.g., Java Card) [56] and reverse engineering techniques [57]. However, since performing a side-channel attack needs special instruments and attack platforms, it is likely that only when the card is in the possession of the attacker for a relatively long period can the data be extracted, in other scenarios the data in the card shall remain confidential. This implies that the common non-tamper-resistance assumption made about the smart cards is *conditional*. As illustrated in Fig. 1 of [7], this conditional non-tamper-resistance assumption about the smart cards has been widely regarded as a basic (also reasonable and prudent) assumption when designing two-factor authentication protocols since 2004.

Naturally, the repeated failure in achieving user anonymity gives rise to an interesting (and foundational) question: *Under the conditional non-tamper-resistance assumption of the smart cards, is it possible to construct a privacy-preserving two-factor authentication scheme for WSNs by only employing symmetric cryptographic techniques?* Besides the aforementioned two-factor schemes for WSNs, dozens of two-factor authentication schemes for various environments (for some latest ones, see [22,58–67]; for a more comprehensive grasp, see those underlined by dash line in Fig. 2 in Section 4.2) also strive to only adopt symmetric-key primitives to reduce the computation overhead while preserving user privacy. It is the main purpose of this paper to demonstrate that, under their non-tamper-resistance assumption of the smart cards, such a strategy is intrinsically infeasible.

Further, as public-key techniques are inherently necessary to achieve user anonymity, another question arises: *How can we design a secure and privacy-preserving two-factor authentication protocol with acceptable efficiency?*

1.3. Our contributions

In this work, we take the first step towards investigating the above two questions and aim to provide definite answers to them. Our contributions lie in the following aspects:

- First, we investigate into two representative schemes, namely Fan et al.'s scheme and Xue et al.'s scheme, to reveal the subtleties and challenges in designing privacy-preserving two-factor authentication schemes for WSNs.
- Second, on the basis of the work of Halevi–Krawczyk (1999) and Impagliazzo–Rudich (1989), we suggest a formal game-based security model for two-factor authentication and put forward a general principle for achieving user anonymity in such schemes, providing a negative answer to the first question.
- Third, though the user anonymity problem is studied in the context of WSNs, we show that our principle can be applied to universal environments, such as the single-server architecture, multi-server environment, global mobility networks, proxy mobile IPv6 networks, satellite networks, mobile clouds and other wireless environments.
- Furthermore, we discuss the viable solutions to implementation issues. By borrowing ideas from privacy-preserving schemes for mobile roaming networks and on the basis of experimental results about public-key primitives, we report that appropriate public-key encryption schemes along with a proper padding mechanism would be promising candidates – providing provable security while attaining reasonable efficiency. This serves as an answer to the second question.

The rest of this paper is organized as follows: in Section 2, we investigate Fan et al.'s scheme. Section 3 describes the privacy flaws of Xue et al.'s scheme. Section 4 devotes to the security model and the principle. Viable solutions are discussed in Section 5, and Section 6 concludes the paper.

2. Cryptanalysis of Fan et al.'s scheme

In 2011, Fan et al. [36] proposed the first denial-of-service (DoS)-resistant and efficient two-factor authentication scheme for WSNs and claimed that their scheme exhibits many merits, such as mutual authentication, user anonymity and local password update, over existing schemes. However, in contrast to their claims, Fan et al.'s scheme is still subject to user anonymity violation attack and other

[2] These three schemes [48–50] only involve symmetric-key cryptographic operations on the mobile user side.

security drawbacks such as vulnerability to smart card security breach attack and insider attack. Here we mainly demonstrate its privacy weakness and the other drawbacks are not our focus. For a self-contained discussion, we first briefly review this scheme.

2.1. Review of Fan et al.'s scheme

There are four entities that participate in the scheme: the user (U_i), the base station (BS), the master node (MN_j) (or called the cluster head) and the sensor node (SN). The base station BS, which generally serves as the gateway node, only takes part in the registration phase and is not involved in the login and authentication process of U_i, MN_j and SN. Following the historical tradition, there are three phases in Fan et al.'s protocol, namely, registration, login and authentication. Before discussing Fan et al.'s scheme, we summarize some intuitive abbreviations that will be used throughout the paper in Table 1, and we will follow the notations in Fan et al.'s scheme as closely as possible.

(1) **Registration phase:** Before the operation of this phase, it is assumed that each master node MN_j, has already established the secret parameter Y_j with the base station BS, and been deployed in the designated area; On the other hand, each sensor node in a cell shares a long term secret number S_k with its master node. Whenever a new user U_i wants to access services from the sensor nodes, she must first register with the base station BS as follows:

Step R1. U_i selects her identity ID_i and password PW_i.

Step R2. $U_i \Rightarrow BS : \{ID_i, PW_i\}$;

Step R3. Upon receiving U_i's registration request, BS chooses a random number R_i and calculates $RID_i = h(R_i \| ID_{BS}) \oplus ID_i \oplus ID_{BS}$, where ID_{BS} stands for the identity of BS. Then BS computes $A_i = h(X) \oplus h^2(ID_i \| PW_i)$ and $V_i = h^3(ID_i \| PW_i)$.

Step R4. BS creates a data table that is composed of the user U_i's warrant N_{ij} (e.g., the master node's identity, the user's access permissions, and the duration of validity) for the relative master nodes and the value of $B_{ij} = h(N_{ij} \| \ Y_j) \oplus h(ID_i \| PW_i)$. BS stores $\{RID_i, R_i\}$ that corresponds to ID_i in its database. Note that the warrant table facilitates fine-grained access control and serves to restrict users as to which cell they can login.

Step R5. $BS \Rightarrow U_i$: A smart card personalized with the security data $\{h(\cdot), RID_i, V_i, A_i\}$, and a table of access warrants $\{N_{ij}, B_{ij}\}$.

(2) **Login phase:** When user U_i wants to obtain real-time information directly from the sensor nodes, she needs to perform the following steps:

Step L1. U_i inserts her smart card into a terminal, and keys her identity ID_i^* and password PW_i^*.

Step L2. The smart card calculates $V_i^* = h^3(ID_i^* \| PW_i^*)$ and checks whether V_i^* is equal to the stored V_i. If they are equal, it implies that ID_i^* equals ID_i and PW_i^* equals PW_i; otherwise, it means that U_i has accidentally input a wrong password or U_i is not a valid card holder, and the smart card refuses to function.

Step L3. U_i selects a master node that she aims to login, and the smart card reads related values of the chosen master node: N_{ij} and B_{ij}. Then, it calculates $TK_i = h((B_{ij} \oplus h(ID_i \| PW_i)) \| T) = h(h(N_{ij} \| Y_j) \| T)$ and $SID_i = RID_i \oplus h((A_i \oplus h^2(ID_i \| PW_i)) \| T) = RID_i \oplus h(h(X) \| T)$, where T denotes the current timestamp on the client side.

Step L4. The smart card computes $C_1 = SID_i \oplus TK_i$ and $C_2 = h(TK_i \| SID_i \| N_{ij} \| T)$.

Step L5. $U_i \rightarrow MN_j : \{N_{ij}, C_1, C_2, T\}$.

Step L6. The smart card erases ID_i, PW_i, $SID_i, h(N_{ij} \| Y_j), h(ID_i \| PW_i)$, and $h^2(ID_i \| PW_i)$ from its memory, but records its last login timestamp T.

(3) **Authentication phase:** This phase aims to achieve mutual authentication among U_i, MN_j and SN. Meanwhile, a session key is established between U_i and SN to secure subsequent data communications. Since this phase has little relevance to our discussions, it is omitted here.

Table 1
Notations and abbreviations.

Symbol	Description
U_i	ith user
BS	Base station
MN_j	jth master node
SN	Sensor node
$\mathcal{A}$	The adversary
ID_i	Identity of user U_i
PW_i	Password of user U_i
X	Secret key maintained by BS
Y_j	Secret parameter shared between MN and BS
N_{ij}	ith user's warrant for jth master node
$\oplus$	The bitwise XOR operation
$\|$	The string concatenation operation
$E(\cdot)/D(\cdot)$	Symmetric encryption/decryption
$h(\cdot)$	Collision free one-way hash function
$\rightarrow$	A common communication channel
$\Rightarrow$	A secure communication channel

2.2. User anonymity violation attack on Fan et al.'s scheme

In this pervasive computing era, individual's sensitive personal information, such as preferences, lifestyles, social circle and residence, may be acquired by an adversary by various means, e.g., through analyzing the session information during a transaction, the services or the resources being accessed [15,68]. As a result, nowadays privacy concerns are rapidly rising among individuals, organizations and governments, privacy-preserving cryptographic protocols are of particular interest. Moreover, in wireless and mobile environments, unauthorized/curious entities may exploit user's static identity to trace the victim's current location and moving history [49,60]. Therefore, user anonymity is a particularly admired property of remote user authentication schemes.

As stated in Section 1, there are a variety of notions of "user anonymity", and quite varied notions may be

implemented in different application environments [19]. As for remote user authentication, user anonymity mainly comprises of two properties, basic and advanced: (1) User identity-protection, which means the adversary could not figure out the real identity of the user; and (2) User untraceability, which guarantees the adversary can neither determine who the user is nor tell apart whether two sessions are executed by the same (unknown) user. The latter (stronger) notion has been widely adopted in most schemes (e.g., [11,23–27]), so does Fan et al.'s scheme.

To achieve user anonymity, a feasible approach is to employ the "dynamic ID technique" which was firstly introduced by Das et al. [69] in 2004: the user's real identity is concealed in the session-variant pseudonym identities. In this way, a user can only be recognized by the authentic server, while all the others over the channel obtain no useful personal information. Schemes that adopt this technique are the so-called "dynamic-ID" schemes [70] or privacy-preserving schemes, and Fan et al.'s scheme falls into this category. However, user anonymity, which is among the crucial aims of Fan et al.'s protocol, cannot be preserved as the following attack demonstrates the failure of their attempt. Note that, this attack has been sketched in a recent work [71], yet for completeness and to illustrate the unique challenges in designing privacy-preserving schemes, here we discuss it in much more detail and mainly focus on its practicality and effectiveness.

Let us see how a dishonest (curious) master node MN_m, who colludes with a malicious privileged user U_m, can successfully breach the un-traceability of any legitimate user, say U_i. The malicious privileged user U_m having her own smart card can gather the parameter A_m from her own smart card, which is realistic when taking into account the state-of-the-art side-channel attacks and reverse engineering techniques [53–55,57]. It has been widely accepted that a desirable two-factor scheme should put aside the tamper-resistance feature of the smart cards when the card is in the possession of the attacker for a relatively long period of time (e.g, several hours or more), and in this case simply assume that the sensitive parameters are extracted [6,32,60].[3] Also note that this assumption has already been explicitly made in Fan et al.'s scheme when they analyze the security of their scheme. With the revealed A_m, U_m can compute $h(X) = A_m \oplus h^2(ID_m \| PW_m)$ because the malicious user U_m knows her own identity ID_m and password PW_m corresponding to her own smart card. With the knowledge of $h(X), U_m$ and MN_m can collude to determine some sensitive user-specific information about the legitimate user U_i by performing the following steps:

Step 1. After receiving U_i's login message $\{N_{im}, C_1, C_2, T\}, MN_m$ computes $TK_i = h(h(N_{im} \| Y_m) \| T)$, where Y_m is pre-shared between MN_m and the base station BS;

Step 2. MN_m computes $SID_i = C_1 \oplus TK_i$;

Step 3. U_m computes $h(X) = A_m \oplus h^2(ID_m \| PW_m)$, where A_m is extracted from U_m's own smart card. Note that the malicious user U_m knows her own identity ID_m and password PW_m corresponding to her own smart card;

Step 4. MN_m colluding with the malicious privileged user U_m now can compute $RID_i = SID_i \oplus h(h(X) \| T)$, where T is intercepted from the public channel.

It is not difficult to see that, RID_i is specific to the legitimate user U_i and kept static in all the login requests of user U_i. Accordingly, this value RID_i can be considered as U_i's "identification", and hence the attacker MN_m can exploit this piece of information to identify and trace the activities of the user U_i, thereby infringing upon U_i's privacy. This contradicts the claim "through clever design, nobody except BS can trace the user U_i" that is made in [36].

There are some points to be noted regarding the above attack. To successfully launch this attack, the dishonest master node MN_m needs to collude with a malicious user U_m. Note that, this assumption is practical, for both the users and the master nodes are not trusted parties and they could be malicious (or at least curious), the collusion of insiders has been widely considered as a practical threat in WSNs [5,73,74]. Although it is also possible that all the master nodes are honest and the users can be trusted, the designers and the users of the scheme should be aware of such a potential weakness. In addition, our attack has nothing to do with the security parameters stored in the victim U_i's smart card. More specifically, MN_m and U_m only need to extract a parameter $h(X)$ from U_m's own smart card to succeed. This assumption is much weaker than revealing secret data from U_i's lost smart card. In a word, to launch the above user anonymity (un-traceability) attack, all that MN_m needs to do is keep an eye over the open channel, while the parameter $h(X)$ can be extracted from an accomplice U_m's smart card in advance. In this regard, our user anonymity violation attack is rather effective and poses a real challenge to two-factor authentication schemes for WSNs.

3. Cryptanalysis of Xue et al.'s scheme

While the attackers colluding together can breach the user anonymity of Fan et al.'s scheme [36], a different attacking strategy has to be taken with Xue et al.'s scheme [27] in place.

For completeness, in this section we briefly review the temporal-credential-based two-factor authentication scheme for WSNs proposed by Xue et al. [27] in 2012. Compared with the earlier schemes such as [11,25,33], Xue et al.'s scheme enjoys a number of important security and useability properties such as mutual authentication, gateway bypassing attack resilience, and friendliness of password update, etc. Overall, its realization is simple and reliable, and thus exhibits great application potential in WSNs. Despite this, we find that it still cannot achieve the property of user anonymity.

[3] It is worth noting that, as deeply investigated in [7,72], smart cards are much more secure than common USB memory sticks, for the former kind of devices is only conditionally non-tamper-resistant, while the latter kind of devices is invariably non-tamper-resistant. As a result, smart-card-based schemes may be used in a malicious card reader, while memory-sticks-based schemes are only fit for the trusted terminals.

3.1. Review of Xue et al.'s scheme

In the following, we employ the notations in Table 1 and will follow the original specifications in Xue et al.'s scheme as closely as possible. This protocol also involves three participants, i.e., the user (U_i), the gateway node (*GWN*) and the sensor node (S_j). Different from Fan et al.'s scheme [36], *GWN* in this scheme not only serves as the registration center but also participates in the authentication process of U_i and S_j. As with most literature in this area, this protocol can be divided into three phases: registration, login, authentication and session key agreement.

(1) **Registration phase:** Before the running of this phase, it is supposed that the identity of the user U_i and the hashed value of her password, i.e. ID_i and $H(PW_i)$, have already been stored on *GWN* side; Each sensor node S_j has its identity SID_j and password PW_j pre-configured, and $H(PW_j)$ stored on *GWN* side. This phase is divided into the user registration and the sensor node registration.

(a) *User registration:* User U_i gets the current timestamp TS_1, and computes $VI_i = H(TS_1 \| H(PW_i))$.

Step RU1. $U_i \rightarrow GWN$: $\{ID_i, TS_1, VI_i\}$.

Step RU2. On receiving the registration request from U_i, *GWN* checks the validity of TS_1. Then, *GWN* computes $VI_i^* = H(TS_1 \| H(PW_i))$ and checks whether $VI_i^* \stackrel{?}{=} VI_i$, where $H(PW_i)$ is extracted from the entry corresponding to ID_i in the back-end database. If it does not hold, *GWN* rejects.

Step RU3. *GWN* further computes $P_i = H(ID_i \| TE_i)$, $TC_i = H(K_{GWN-U}) \| P_i \| TE_i$ and $PTC_i = TC_i \oplus H(PW_i)$, where TE_i is the expiration time of the temporal credential set by *GWN* or the trusted third party (TTP), K_{GWN-U} is *GWN*'s private key and TC_i is the temporal credential for U_i issued by *GWN*. At last, *GWN* personalizes the smart card for U_i with the parameters $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, PTC_i\}$.

Step RU4. *GWN* $\rightarrow U_i$: A smart card containing security parameters $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, TC_i\}$.

(b) *Sensor node registration:* As this phase does not pertain to our discussions, it is omitted.

(2) **Login phase:** In this phase, U_i constructs her login request and sends it out to *GWN*:

Step L1. U_i inserts her smart card into a card reader and inputs her identity ID_i^* and password PW_i^*.

Step L2. The smart card verifies whether the input ID_i^* equals the stored ID_i and whether $h(PW_i^*)$ equals the stored $h(PW_i)$. If both verifications hold, it indicates that U_i is a legal card holder. Then, the smart card computes $TC_i = PTC_i \oplus H(PW_i)$.

Step L3. U_i obtains the current timestamp TS_4 and selects a random number K_i. Then U_i computes $DID_i = ID_i \oplus H(TC_i \| TS_4)$, $C_i = H(H(ID_i \| TS_4) \oplus TC_i)$ and $PKS_i = K_i \oplus H(TC_i \| TS_4 \| \text{“000”})$. It should be noted that $H(TC_i \| TS_4 \| \text{“000”})$ is different from $H(TC_i \| TS_4)$, which is ensured by the target-collusion-resistant nature of Hash functions.

Step L4. $U_i \rightarrow GWN$: $\{DID_i, C_i, PKS_i, TS_4, TE_i, P_i\}$.

(3) **Authentication and session key agreement phase:** This phase aims to achieve the goal of mutual authentication among U_i, *GWN* and S_j. Meanwhile, a session key is negotiated between U_i and S_j. We skip the review of this phase as it has little relevance regarding our discussions.

3.2. Cryptanalysis of Xue et al.'s scheme

In Xue et al.'s scheme [27], one assumption made about the capabilities of the adversary $\mathcal{A}$ is that it has total control over the communication channel among the user U_i, the gateway node *GWN* and the sensor node S_j, which is consistent with the common adversary model for distributed computing, called the standard model or the Dolev–Yao model [75]. Under this assumption, $\mathcal{A}$ can intercept, block, delete, insert or alter any messages exchanged in the channel. Nevertheless, $\mathcal{A}$ is restricted from breaking the cryptographic primitives (e.g., hash functions, block ciphers), that is, the computational capability of $\mathcal{A}$ is powerful but not omnipotent. Another assumption explicitly made in this scheme is that smart cards can be tampered. However, in the following discussion of the privacy vulnerability of Xue et al.'s scheme, we will show how an adversary $\mathcal{A}$ only with the first assumption made (i.e., $\mathcal{A}$ has total control over the open channel) can successfully breach the claimed goal of user anonymity:

Step 1. Intercepts a login request message, say $\{DID_i, C_i, PKS_i, TS_4, TE_i, P_i\}$, sent by U_i;

Step 2. Guesses the value of ID_i to be ID_i^* from a dictionary space $\mathcal{D}_{id}$.

Step 3. Computes $P_i^* = h(ID_i^* \| TE_i)$, where TE_i is intercepted as in Step 1.

Step 4. Verifies the correctness of ID_i^* by checking if the computed P_i^* equals the intercepted P_i.

Step 5. Repeats the above Steps $2 \sim 4$ until the correct value of ID_i is found.

To launch the above attack, $\mathcal{A}$ only needs to eavesdrop a single run of the protocol and then she offline guesses the victim's identity without interacting with *S*, and there is no special or expensive operations involved, such as power analysis or reverse engineering. Let $|\mathcal{D}_{id}|$ denote the number of identities in $\mathcal{D}_{id}$. The running time of the above attack procedure is $\mathcal{O}(|\mathcal{D}_{id}| * T_H)$, where T_H is the time for Hash operation. So, the time for $\mathcal{A}$ to recover U_i's identity is a linear function of $|\mathcal{D}_{id}|$.

3.3. Effectiveness of our attack

To gain an intuitive grasp of the effectiveness of the above attack, we further obtain the running time for cryptographic operations using the publicly-available crypto-

graphic library MIRACL [76], a multi-precision and rational arithmetic C/C++ library. We carry out experiments on Laptop PCs with different computational power. For accuracy, each operation is repeated for a thousand times. Table 2 lists the experimental data for related operations on common Laptops.

As $|\mathcal{D}_{id}|$ is very limited in practice, e.g. $|\mathcal{D}_{id}| \leqslant |\mathcal{D}_{pw}| \leqslant 10^6$ [77–79], the above attack can be completed in polynomial time. Note that, in this user-identity breach attack, the adversary only needs to keep an eye over the public channel and it does not involve any special cryptographic operations (e.g., side-channel analysis). Consequently, the proposed attack is indeed practical, and $\mathcal{A}$ may recover the identity in seconds on a laptop PC.

4. The public-key principle for two-factor authentication with user anonymity

Since sensor nodes and smart cards are extremely resource-constrained devices with low battery power, low computation capability and limited memory capacity, protocol designers are faced with the hard task of reconciling security, efficiency and functionality requirements, and often must make design decisions which are seemingly well motivated but may have unintended consequences. Without some necessary design principles (guidelines), the designers can only try their best to ensure the protocol is infallible, which would help explain why in practice most of the complex security protocols have eventually been revealed to have weaknesses to be fixed [80–82].

It is widely recognized that the certificate-based authentication schemes are too bulky and complex for WSNs, and traditional schemes (e.g., [83,84] designed for the Internet) using public-key operations (e.g., modular exponentiation, Elliptic Curve point multiplication and Pairing) are also computation intensive and power hungry. Unsurprisingly, most two-factor authentication schemes for WSNs (e.g., [11,23–25,27,30,33,35,36,51,52]) swing to the other extreme: they attempt to only adopt non-public-key techniques (e.g., Hash functions, symmetric encryptions, XOR operations, MAC operations and pseudorandom functions) to reduce the computational complexity, communication cost and storage overhead while fulfilling the stringent security and privacy requirements, but in vain. To explicate this repeated but rather obscure failure and to overcome the perennial problem, in this section we put forward a general principle which would serve as a primary guidance to prevent common errors in future designs of privacy-preserving schemes.

4.1. The public-key principle for user anonymity

We have analyzed more than one hundred recently proposed two-factor authentication schemes for general purpose, more than thirty schemes for wireless communications and twenty-seven schemes for WSNs, and some of our recent cryptanalysis results include [7,21,71,83,85–87]. We observed that schemes which do not employ public-key techniques are invariably unable to achieve user anonymity, if the smart cards are assumed to be (conditionally) non-tamper resistant when lost. In other words, all these schemes that only employ non-public-key techniques but purport to preserve user privacy are problematic, and some quite recent examples include [22,26,27,30,36,51,58–60,62–64,66,88–91]. In what follows, based on the work of Halevi–Krawczyk [44] and Impagliazzo–Rudich [43], we prove that this is no accident: Under the (conditional) non-tamper-resistance assumption of the smart cards, their strategy for preserving privacy is intrinsically infeasible.

(1) Halevi–Krawczyk model for password authentication

In [44], Halevi and Krawczyk proposed a password-based authentication protocol and defined a security model for this type of protocols (i.e., one-factor). This model is built on the standard adversary model for distributed computing [75], and focuses on user-to-server one-way authentication. Since, by their nature, passwords are drawn from a finite set of possibilities, an adversary can always select a trial password from the relatively small password space and then honestly follow the protocol to launch an on-line impersonation attack, which can be thwarted in practice by locking a user's account after a certain number of consecutive authentication failures. Informally, a password-based one-way authentication protocol is said to be secure if such an on-line guessing attack is the "best" strategy for the adversary to compromise user authentication.

Following the common practice in cryptography, the Halevi–Krawczyk security model [44] for password-based one-way authentication is defined by a "probabilistic game", mainly including the players and rules of the game, the attacker and its capabilities, and the notion of security:

Definition 1 (*Parities*). Parties in the game are client U, server S and the adversary $\mathcal{A}$.

Definition 2 (*Game rules*). The game is parameterized by a security parameter k and a public password dictionary $\mathcal{D}$ of small size. The game proceeds as follows:

- *Set-up phase:* the server S picks a name, which is an arbitrary string denoted by ID_S, and publishes it. S also chooses its cryptographic keys and publishes its public key (if any); U picks an arbitrary string ID_i as her name and uniformly draw a password PW_i from $\mathcal{D}$,[4] and gives PW_i to S while keeping it secret from $\mathcal{A}$. $\mathcal{A}$ can also register additional users with S at any time (before, during, or after the set-up phase) by picking any pair of user name AID_i and password APW_i (provided that $AID_i \neq ID_i$), publishing AID_i and giving APW_i to S.
- *Game running phase:* $\mathcal{A}$ has full control over the communication channel between the user U and server S. In other words, every message that U and S send goes to $\mathcal{A}$, and every message they receive is from $\mathcal{A}$. $\mathcal{A}$ can intercept, block, delete, insert or alter any messages exchanged in the channel. Moreover, $\mathcal{A}$ may also initiate

[4] Everything would work with any other distribution over $\mathcal{D}$ [92], the uniform-distribution assumption is just to make expressions simpler.

Table 2
Computation evaluation (in microseconds) of related operations on common Laptop PCs.

Experimental platform (Laptops) (GHz)	Hash operation T_H (SHA-1) (μs)	Other lightweight operations (e.g., XOR) (μs)
Intel T5870 2.00	2.437	0.011
Intel i5-3210 2.50	1.106	0.009
Intel i3-530 2.93	0.815	0.008

new sessions by sending special "prompt" messages to the corresponding parties. This game is run until $\mathcal{A}$ decides to halt.

Definition 3 (*Feasible adversary*). The capabilities of the adversary $\mathcal{A}$ are modeled through the above game rules and measured in computational "feasibility", e.g. probabilistic polynomial time (PPT), and are summarized as follows:

(i) $\mathcal{A}$ can control the messages exchanged in the channel between the user U and server S.
(ii) $\mathcal{A}$ can offline enumerate the password space $\mathcal{D}$.
(iii) $\mathcal{A}$ has the ability to create and register additional users. That is, $\mathcal{A}$ may be a legitimate but malicious user.

However, $\mathcal{A}$ is restricted from breaking the cryptographic primitives (e.g., encryption, Hash). As the computational power of $\mathcal{A}$ is large but limited, it is called a feasible attacker.

For simplicity and clarity, before presenting the theoretical definition of security for the model, i.e. Definition 10, we first give some preliminary terminologies.

Definition 4 (*Output pair*). Each authentication session has a unique session identifier *sid*, and each party is required to record events related to the security of the authentication. Namely, at the end of each session between U and S, user U outputs a pair (S, sid); server S outputs a pair (U, sid). If (alleged) user U during the session *sid* fails to authenticate itself to server S, S outputs $(U, sid, \perp)$. Without loss of generality, *sid* is set to be the (ordered) concatenation of all messages sent and received by the corresponding party.

Definition 5 (*Correctness*). A protocol is said to be syntactically correct if whenever all the messages between U and S in session *sid* are passed unchanged, then S and U output (U, sid) and (S, sid), respectively. That is, at the end they both accept (and conclude with the same session key).

Definition 6 (*Impersonation*). An event in which the server outputs (U, sid) but the intended user U never outputs a pair (S, sid) is called a successful impersonation. An event in which the server outputs a pair (U', sid) after already outputting some other pair (U'', sid) , is called a successful replay. An event in which the server outputs $(U, sid, \perp)$ is called an authentication failure.

Definition 7 ((ℓ, m)-*win*). An (ℓ, m)-run of the game is a run in which $\mathcal{A}$ performs at most m active impersonation attempts, and U outputs at most ℓ pairs of (S, sid). $\mathcal{A}$ is said to have achieved an (ℓ, m)-win if in an (ℓ, m)-run of the game, there is at least one successful impersonation event.

Definition 8 (*Secure U2S password authentication*). Let $\epsilon(\cdot, \cdot, \cdot)$ be a positive real function and let $\mathcal{P}$ be a syntactically correct password one-way (i.e., user to server, or *U2S* for short) authentication protocol. We say that $\mathcal{P}$ ensures *U2S* authentication up to ϵ, if for any feasible adversary $\mathcal{A}$, any finite dictionary $\mathcal{D}$, any sufficiently large security parameter k, any polynomial ℓ, and any integer $m < |\mathcal{D}|$, we have

$$\Pr[(\ell, m) - \text{win}] \leqslant \frac{m}{|\mathcal{D}|} + \epsilon(k, \ell, m)$$

where ℓ is the upper-bound of user output pairs and m the upper-bound of active impersonation attempts made by $\mathcal{A}$.

(2) Proposed security model for two-factor authentication

Based on the Halevi–Krawczyk security model [44] for password authentication, we proceed to define a security model for two-factor authentication. As with [44], here we restrict ourselves to one-way authentication and key exchange for simplicity. It is not difficult to see that security notions of more general tasks, such as two-way authentication, can be formalized in a similar way. We only take into consideration of the most general case where there is a user U and a server S involved in the protocol. Note that, our model can be applied to specific applications where there may be more parties other than U and S. For example, in the scenario of WSNs [11], the master node and sensor node together can be seen as the server, while in the context of mobile networks [49] the home agent and foreign agent together are seen as the server.

The players in our security game are the same with that of the Halevi–Krawczyk game. The main differences in game rules between the two games are: (1) in the set-up phase, S additionally issues a smart card with some parameters to U; and (2) in the running phase, $\mathcal{A}$ can further extract security parameters from a lost (stolen) smart card, or from a valid smart card owned (registered) by $\mathcal{A}$ herself.

Definition 9 (*Adversary for two-factor authentication*). According to our game rules, the adversary in our security model is equipped with the Capabilities i–iii (see Definition 3) as well as the following capability:

(iv) $\mathcal{A}$ can reveal the secret parameters stored in U's card when $\mathcal{A}$ somehow (e.g. picking up, stealing) gets access to it for a relatively long time. This implies that $\mathcal{A}$ can also extract data from the smart cards registered by herself.

As justified in [6,93], equipping an adversary with the Capabilities i–iv is indeed reasonable for two-factor authentication schemes, and this practice has gained general acceptance (e.g., [6,7,32,33,35,36,51]).

As password authentication constitutes the basis of two-factor authentication, the preliminary terminologies such as "Output pair", "Correctness", "Impersonation" and "(ℓ, m)-win" defined in the Halevi–Krawczyk security model (for password authentication) can be directly adopted into the definition of security for two-factor authentication. Based on the above game and terminologies, we proceed to define what it means to be a secure two-factor authentication protocol. It is obvious that if U's password and smart card are both compromised, there is no way to prevent $\mathcal{A}$ from masquerading as U, and any two-factor authentication protocol cannot survive such a trivial attack.

Consequently, there are three cases left: (1) both U's password and smart card remain secure, in this case the protocol, of course, remains secure; (2) $\mathcal{A}$ has compromised U's password but not the smart card, in this case since there are often high entropy cryptographic keys stored in the card, $\mathcal{A}$ will pose no practical danger to the protocol without the U's smart card; and (3) $\mathcal{A}$ has compromised U's smart card but not the password, in this case the only defense line now lies on the low entropy password, which is rather dangerous because the two-factor scheme now is downgraded to a traditional password only protocol. This case is the best chance for $\mathcal{A}$ to break the protocol.*Therefore, we say a two-factor authentication is secure (informally) if $\mathcal{A}$ cannot succeed with a non-negligible probability in the third case*. This well explains why the notion of security in our model is the same with that of the Halevi–Krawczyk model. To deal with privacy-preserving protocols, we also formalize the notion of user anonymity.

Definition 10 (*Secure U2S two-factor authentication*). Let $\epsilon(\cdot,\cdot,\cdot)$ be a positive real function and let $\mathcal{T}$ be a syntactically correct two-factor one-way (i.e. user to server, or *U2S* for short) authentication protocol. We say that $\mathcal{T}$ ensures *U2S* authentication up to ϵ, if for any feasible adversary $\mathcal{A}$, any finite dictionary $\mathcal{D}$, any sufficiently large security parameter k, any polynomial ℓ, and any integer $m < |\mathcal{D}|$, we have

$$\Pr[(\ell, m) - \text{win}] \leqslant \frac{m}{|\mathcal{D}|} + \epsilon(k, \ell, m)$$

where ℓ stands for the upper-bound of user output pairs and m the upper-bound of active impersonation attempts made by $\mathcal{A}$ in our security game.

Definition 11 (*User anonymity/un-traceability*). As user identities are self-chosen memorable strings, usually subject to a pre-defined format and less protected than the passwords (e.g., displayed in plain-text and their entire list often publicly available), $\mathcal{A}$ can enumerate the user identity space $\mathcal{D}_{id} = \{ID_1, ID_2, \cdots, ID_N\}$, where $N \geqslant 2$. But even in such situations, what a protocol $\mathcal{T}$ with user anonymity can ensure is that, based on the protocol transcripts, $\mathcal{A}$ still cannot decide: (1) which user it is in a given login session (e.g., session sid_j); and (2) whether a specific user (e.g., user U_i) has been involved in two (or any number of) given sessions.

(3) Proof of the public-key principle

Prior to Havelli–Crawczyk's work, Impagliazzo and Rudich [43] devised a security model in which "only symmetric-key tools are available" and they investigated whether a specific kind of protocols can be obtained in this model. There are two features that distinguish the Impagliazzo–Rudich model from the standard model of distributed computing:

- The adversary is given an oracle access to an NP-complete problem. Particularly, this implies that given a "public key", the adversary is able to find the corresponding "private key" in polynomial time and all public-key cryptographic primitives are no longer existent.
- All the parties are granted access to a random function f, which maps each string in $\{0,1\}^*$ into a uniformly distributed k-bit string, where k is a security parameter. Specifically, this means cryptographic primitives, such as collision-intractable hashing, MAC and symmetric encryption, can be easily implemented using f.

Impagliazzo–Rudich then reported that secure key-exchange (in a broad sense) protocols are impossible in this model:

Lemma 1 (*Impagliazzo–Rudich 1989*). *There is no key-exchange protocol, in the Impagliazzo–Rudich model, which is secure up to $\epsilon'(k)$, for any $\epsilon'(k) \leqslant \frac{1}{2^k}$, where k is the system security parameter and $\epsilon'(k) = \epsilon(k, 1, 1)$.*

Using this lemma, we can reach the following conclusion:

Theorem 1. *Under the (conditional) non-tamper-resistance assumption about the smart cards, there is no smart-card-based password protocol with user anonymity that can ensure one-way authentication up to $\epsilon(k, \ell, m)$ by only using symmetric cryptographic primitives, for any $\epsilon(k, \ell, m) \leqslant \ell \cdot m \cdot \epsilon'(k) \leqslant \frac{\ell m}{2^k}$, where k is the system security parameter, ℓ the upper-bound of user output pairs and m the upper-bound of active impersonation attempts made by $\mathcal{A}$ in the security game.*

Proof. The proof is by contradiction. Its basic idea is that, given a two-factor authentication protocol that supports user anonymity, we can use it (without any further cryptographic primitives) to implement a secure key-exchange protocol, while the latter cannot be built on mere symmetric-key primitives, so is the former.

To get the contradiction we assume that, under the Capabilities i–iv of the adversary (hereafter we call it "our adversary model", which is also the most widely adopted model in such schemes), there exists a smart-card-based password protocol $\mathcal{T}$ with user anonymity which only employs symmetric cryptographic primitives. It is crucial to notice that, in our model, all the security parameters stored in the smart card can be extracted by side-channel attacks or reverse engineering techniques [53–56]. That is, once the security of the smart card is breached, the security of $\mathcal{T}$ is completely dependent on the security of the password. And hence, protocol $\mathcal{T}$ now is downgraded to a traditional one-factor password authentication scheme, we called it protocol $\mathcal{P}$.

The underlying adversary model of password protocol $\mathcal{P}$ is actually the Impagliazzo–Rudich model because $\mathcal{P}$ is required to only employ symmetric cryptographic primi-

tives, which is consistent with this model but not the standard model. As $\mathcal{T}$ is assumed to be a two-factor authentication protocol with user anonymity in our model, now $\mathcal{P}$ is a one-factor authentication protocol with user anonymity in the Impagliazzo–Rudich model, which means $\mathcal{A}$ cannot distinguish whether two sessions originate from the same user in a protocol run of password protocol $\mathcal{P}$.

Then we show password protocol $\mathcal{P}$ can be exploited to construct a one-factor (i.e., password-based) key exchange protocol which is secure up to $\epsilon'(k)$ in the Impagliazzo–Rudich model. Firstly, we alter the way $\mathcal{P}$ generate the session key. In each session, say sid_j, every party in the protocol execution (i.e., U_i and S) can exchange one bit b_j by setting $b_j = h(sid_j \| ID_i \| ID_S) \bmod 2$, where $h(\cdot)$ is a target collision-resistant hash function instantiated by the random function f which is defined in the Impagliazzo–Rudich model. At the end of the successful protocol run, U_i and S indeed can compute the same bit b_j, for they know the right relation between sid_j, ID_i and ID_S. Now we show that $\mathcal{P}$ can be used to implement the exchange of longer keys, e.g. $l(\geqslant \log_2 k)$ bits, by simply repeating the same protocol l times and setting a cryptographic key $sk = \overbrace{b_j \ldots b_m \ldots b_t}^{l\text{bits}}$, where b_m and b_t are two bits exchanged (in the same way with b_j) in session sid_m and sid_t, respectively. We call this resulting key exchange protocol $\mathcal{E}$.

In the key exchange protocol $\mathcal{E}$, firstly, it is evident that U_i and S will successfully exchange the same bit-string sk at the end of l sessions. Secondly, we consider an augmented adversary $\mathcal{A}^+$, which possesses all the abilities of $\mathcal{A}$ and is additionally equipped with the capability of determining whether two different sessions are originated from the same (but unknown) user. Now, $\mathcal{A}^+$ can correctly compute the bit-string sk that protocol $\mathcal{E}$ outputs with probability $\frac{1}{|\mathcal{D}_{id}|}(\gg \epsilon'(k))$ as follows: (1) Selects the l sessions, say $sid_j, \ldots, sid_m, \ldots sid_t$, in which the target user has involved[5]; and (2) Guesses the target user's identity to be ID_i' from the identity space $\mathcal{D}_{id}$; (3) Computes $sk' = b_j' \| \ldots \| b_m' \| \ldots \| b_t'$, where $b_j' = h(sid_j \| ID_i' \| ID_S) \bmod 2, b_m' = h(sid_m \| ID_i' \| ID_S) \bmod 2$ and $b_t' = h(sid_t \| ID_i' \| ID_S) \bmod 2$. It is crucial to note that, for any adversary (i.e., $\mathcal{A}$ or $\mathcal{A}^+$), to compute (but not simply guess) sk, she must correctly select the l sessions in which U_i has indeed involved. Without such a correct selection, success, of course, is impossible. At this point, it is not difficult to see that, this is the most effective way for the augmented adversary $\mathcal{A}^+$ to compute the key sk' that is exchanged in protocol $\mathcal{E}$.

Thirdly, since protocol $\mathcal{P}$ achieves user un-traceability, $\mathcal{A}$ is not able to figure out whether two sessions are originated from the same (but unknown) user. That is, given a sequence of $x(x \geqslant l)$ sessions, the probability for $\mathcal{A}$ to select l sessions in which U_i has exactly involved is no larger than $1/\mathbf{C}_x^l$, where $\mathbf{C}_x^l$ denotes the combination numbers of l out of x. Accordingly, the probability $\mathrm{Adv}_{\mathcal{P}}^{\mathrm{sk}}(\mathcal{A})$ for $\mathcal{A}$ to compute the correct session key sk is also no larger than $1/\mathbf{C}_x^l$. In sum, $\mathrm{Adv}_{\mathcal{P}}^{\mathrm{sk}}(\mathcal{A}) \leqslant \frac{1}{2^l} \leqslant \epsilon'(k)$, when $x \gg l$, e.g. $x = 2l, l \geqslant k = 160$. Consequently, protocol $\mathcal{P}$ can be used to construct a key exchange protocol $\mathcal{E}$ which is secure up to $\epsilon'(k)$, though the performance of $\mathcal{E}$ is not so satisfactory.

On the other hand, according to Lemma 1, no key exchange protocol can be secure up to $\epsilon'(k)$ in the Impagliazzo–Rudich model in which only symmetric-key techniques are provided (in other words, all "public key tools" are eliminated), and $\mathcal{E}$ is no exception. At this point, the paradox arises. This indicates that the user-anonymity assumption made about the protocol $\mathcal{P}$, which is the cornerstone of $\mathcal{E}$, does not hold. It further invalidates the user-anonymity assumption made about the two-factor scheme $\mathcal{T}$, which completes the proof. □

By following this principle, one can easily affirm that, besides Xue et al.'s scheme [27] and Fan et al.'s scheme [36] that we just have scrutinized, all these newly proposed two-factor schemes for WSNs [25,26,35,51,52], which attempt to meet stringent security and privacy requirements while using only lightweight symmetric cryptographic primitives (e.g., hash functions, MACs, block ciphers and exclusive-OR operations), fail to achieve user anonymity. For all we know, most of these schemes were just made online and have not been cryptanalzed elsewhere. Some of them, like [35,36,52], even have been provided with a formal security proof. Nonetheless, they are inherently unable to preserve privacy due to the violation of this basic principle.

4.2. Universal applicability of our principle

Remarkably, our principle has universal applicability for privacy-preserving two-factor authentication schemes for general environments like single-server architecture [64,89], multi-server environment [22,70], traditional mobile networks [59,60], Mobile IPv6 networks [91], the global mobility networks [86,94] and the satellite networks [66], etc. The main reason is that the underlying adversary model (i.e. "adversary for two-factor authentication", see Definition 9) on which our principle relies, can be directly applied to the above-mentioned application scenarios. More specifically, with the same assumptions about adversary capabilities and the same definitions of privacy goals (i.e., user identity protection and user un-traceability), the same class of fundamental cryptographic tools (i.e., symmetric-key techniques or public-key techniques) shall be put in use, no matter for which specific environment a scheme is designed.

By following our principle, one can find that all these two-factor schemes (as *underlined by dash line* in Fig. 2, including these well-cryptanalyzed ones [49,60,61,64,70,95,96] and these newly proposed ones [22,58,59,62,65–67,89–91,97–99], to name just a few), which only employ symmetric-key techniques on the mobile user side, but purport to preserve user anonymity, are definitely problematic just at a single glance. Note that many other important schemes cannot be included into Fig. 2 only due to space constraints. In the following we only take Chuang et al.'s scheme [91] as a demonstration

[5] Note that once the l sessions have been selected, they can be rearranged in chronological order (e.g., according to their start time).

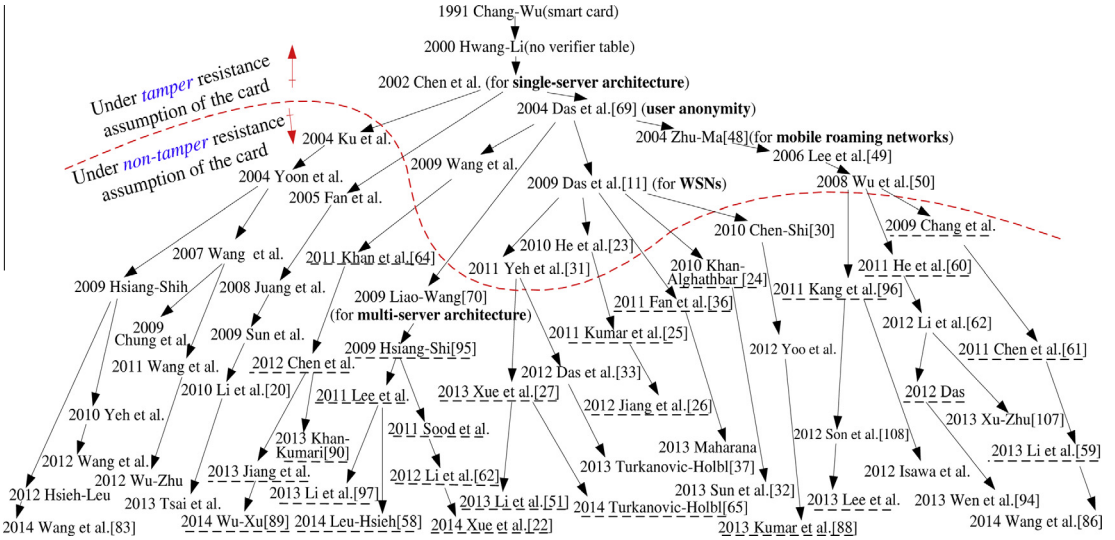

Fig. 2. A brief history of privacy-preserving two-factor authentication for various environments.

of the universal applicability of our principle, and interested readers can confirm themselves that any of these newly developed schemes (e.g., [22,58,5 9,62,65,67,89,9 0,97–99]) would equally serve as an ideal exemplum and could somehow be breached.

Review of Chuang et al.'s scheme [91]: In 2013, Chunag et al. propose a privacy-preserving two-factor authentication scheme for protecting mobile users and other network entities (including the mobile access gateway (MAG), the local mobility anchor (LMA) and the authentication, authorization and accounting (AAA) server) from malicious attacks in Proxy Mobile IPv6 networks. This scheme uses similar notations to Fan et al.'s scheme (see Table 1), and here we only review the critical parts relevant to our discussion, while the other details are omitted. This scheme has three phases: registration, authentication and password change. In the registration phase, the mobile user MN submits her identity ID_{MN} and password ID_{MN} to the AAA server, then AAA server stores the parameters $\{ID_{MN}, c_1 = h(ID_{MN}\|sv), c_2 = h(PW_{MN}) \oplus c_1, c_3 = E_{PSK}(ID_{AAA}\|sv), c_4 = h(ID_{AAA}\|sv),\ c_5 = h(sv), h(\cdot)\}$ on a smart card and issues it to MN through a secure channel, where ID_{AAA} and sv stand for AAA's identity and long-term secret key, respectively. In the authentication phase, MN only directly interacts with LMA. First, MN inserts her card into a terminal, and keys ID_{MN} and PW_{MN}. Then, smart card verifies the correctness of ID_{MN} and PW_{MN}, chooses a random N_1, computes $AID_{MN} = ID_{MN} \oplus h(c_5\|N_1)$ and $Auth_{MN} = h(c_1\|N_1)$, and then sends $\{AID_{MN}, c_3, E_{c_4}(Auth_{MN}\|N_1)\}$ as her login request to LMA. The rest of the protocol specifications have little relevance to our discussion and thus are omitted.

Cryptanalysis of Chuang et al.'s scheme [91]: Suppose a legitimate but dishonest user $MN_{\mathcal{A}}$ who has registered with AAA, which is also the registration center of many other users like an innocent MN. $MN_{\mathcal{A}}$ can extract the parameters $\{c_4, c_5\}$ from her own smart card. Note that, the parameters c_4 and c_5 are identical in all users' (including MN's) smart cards. The assumption of extracting data from smart cards is quite practical when considering the state-of-the-art side-channel attacks and reverse engineering techniques [53–57,100], and actually it has been widely accepted as a basic assumption when designing smart-card-based protocols since 2008 [6,101,102]. With c_4 and c_5 revealed, $MN_{\mathcal{A}}$ can disclose MN's identity ID_{MN} as follows:

Step 1. Intercepts a login request message, say $\{AID_{MN}, c_3, E_{c_4}(Auth_{MN}\|N_1)\}$, sent by MN;

Step 2. Decrypts $E_{c_4}(Auth_{MN}\|N_1)$ using c_4 to get $Auth_{MN}\|N_1$, where c_4 is extracted from $MN_{\mathcal{A}}$'s own smart card and this value is identical to c_4 that is stored in MN's smart card.

Step 3. Computes $ID_{MN} = AID_{MN} \oplus h(c_5\|N_1)$, where c_5 is extracted from $MN_{\mathcal{A}}$'s own smart card and this value is identical to c_5 stored in MN's card.

The above attack reveals that, once the secret parameters stored in the adversary $MN_{\mathcal{A}}$'s own smart card are extracted, any other legitimate user's identity can be obtained by simply eavesdropping the open (wireless) communication channel. It is worth noting that, $MN_{\mathcal{A}}$ only needs to extract the secret data from her own card but not the victim's lost card, and she may turn to professional organizations (labs) for help. In this light, our attack is rather realistic. Obviously, the disclosure of mobile user MN's identity ID_{MN} may facilitate tracking of MN's activities like the moving history, lifestyles, even her current location [103]. This well explains why user anonymity is an important feature that security protocols for mobile computing environments should provide. In a word, our analysis well demonstrates the feasibility of user privacy breach attack on Chuang et al.'s scheme [91], thereby contradicting the claim made in [91] and illustrating the universal applicability of our principle.

5. Potential countermeasures

Having proved our principle and demonstrated its universal applicability, we proceed to discuss the corresponding countermeasures. Since symmetric-key techniques are not sufficient to achieve user anonymity (un-traceability), the available choice is obvious – resorting to public-key techniques. Now, the remaining key issue is how to integrate public-key primitives into traditional only symmetric-key based schemes.

5.1. Pre-loading a pseudo-IDs pool

An obvious way is to pre-load a large pool of pseudonym identities, which are signed by the server's private key, in a user's smart card, and then the user uses a fresh pseudo-ID each time to login to the server. However, such an idea may be suitable for the environments where the users generally have large storage capacity [104], but will not work well in smart-card-based schemes where the mobile user's storage capacity is particularly limited. What's more, once a victim's mobile device (smart cards) is lost and these pseudonym identities are revealed, the entire system will be endangered.

5.2. Group and ring signatures

In fact, user anonymity was first addressed by the introduction of group signatures [105]. A group signature scheme allows a group member to sign on behalf of the whole group anonymously without disclosing her own identity, and the verifier of the group signature could not tell who is the real signer in the group. User anonymity can also be realized using the ring signature technique [106]. In a ring signature scheme, a signer takes the public-keys of other ring members as input to sign a message without the assistance or even awareness of the other members, and the generated ring signature can convince any verifier (including the group manager) that the message was indeed signed by one of the ring members without revealing the signer's identity.

Both group and ring signatures share the same idea of achieving user anonymity – hiding one's own identity among a group of identities. The main differences between them are that: (1) the group is formed more freely in the ring signature; and (2) unlike the ring signature, the group members in the group signature only look indistinguishable to the verifier (but not to the group manager). In other words, each has its own attractive characteristics for anonymous user authentication. However, they could not be readily used in two-factor authentication schemes, because each member shall be equipped with a certificate and thus they cannot work without the support of Public Key Infrastructure (PKI). What's more, in existing ring or group signature schemes, the computation cost often increases linearly with the group or ring size, which is undesirable for two-factor systems especially when the number of users is large. As a result, in its current form, neither of them is practical for implementation in wireless mobile applications.

5.3. IND-CCA2 public-key encryption

Luckily, we find several latest schemes [83,84,102,107,108], which adopt a different method from the above-mentioned techniques, manage to preserve user anonymity while not largely impairing efficiency and not sacrificing security assurance and desirable functionalities. Some of them like [83,102,107] even have achieved provable security.

These four schemes are designed for different application environments (not including WSNs), but they have one thing in common – the server possesses a pair of private and public keys, while each user holds a smart card personalized with the server's public key, memorizes a weak password and implicitly encrypts her real identity with the help of the server's public key when constructing login request. In this way, user identity-protection (and un-traceability) is attained on the basis of the intractability of well-known **NP** problems such as the integer factorization problem (IFP), discrete logarithm problem (DLP), computational Diffie–Hellman problem (CDHP) and their variants like quadratic residue problem (QRP) and bilinear computational Diffie–Hellman problem (BCDHP). However, in these schemes, no rationale or justification has been provided for preferring these comparatively costly public-key techniques rather than lightweight symmetric-key primitives. Fortunately, in the above section we have at last settled this fundamental issue.

In addition, the discussion on the theory behind the design choices of these schemes [83,84,102,107,108] is not sufficient, either. It would be better if more emphases had been placed on the design philosophy, rather than the mere statement that such-and-such a protocol attains user anonymity. To further clarify this, we take the scheme in [102] for a concrete example. This privacy-preserving scheme is advanced by Zhou–Xu in 2011 and aimed for roaming service in global mobility networks. The authors first presented the scheme and then gave a formal (intricate yet quite routine) security proof, but no basic idea about how and why their scheme can achieve user anonymity is provided. Here we try to shed some light on it. In Zhou–Xu's scheme, the user U_i's pseudonym identity SID_i is encrypted using a variant of the Elgamal algorithm: setting $SID_i = ID_i \oplus h(D\|n_i)$, where $D = B^a \bmod p, a$ is a random number chosen by U_i herself, $B(= g^b \bmod p)$ is the home server's public key and g is a generator of a large group with prime order p. U_i sends out $\{SID_i, n_i, A = g^a \bmod p, V = h(C\|D), ID_H\}$ as her login request, where C is U_i's static credential issued by the home server and ID_H is the home server's identity.

If we are not mistaken, the underlying rationale behind Zhou–Xu's scheme [102] is that, without the ability of computing $D = g^{ab}$ from the Diffie–Hellman triple (g, g^a, g^b) , no one other than U_i and its home server can compute D to recover the real identity ID_i from the pseudonym SID_i. For the same reason, no one other than U_i and its home server can compute the user' dynamic credential V to impersonate U_i. Due to the dynamically changed nature of D, all the elements in the login request that may be exploited to derive some specific information about U_i are made session-variant, and thus user un-traceability is

Table 3
Execution time (in seconds) for typical public-key operations on sensor nodes and smart cards.

Experimental platform	RSA encryption/decryption ($\|n\| = 1024, e = 2^{16} + 1$)	Modular exponentiation ($\|n\| = 512$)	Point multiplication ($E(\mathbb{F}_p), \|p\| = 160$)	Tate pairing ($E(\mathbb{F}_p), \|p\| = 1024$)
Mica2 ATmega128L 8-bit 4 MHz	0.862/22.029	2.679	1.616	45.632
MicaZ ATmega128 8-bit 8 MHz	0.430/10.990	5.370	0.810	22.765
Philips HiPerSmart 32-bit 36 MHz	0.006/ 0.140	0.068	0.011	0.290

preserved. Exactly the same approach – concealing both the user identity and long-term credential in a single encryption – is taken in [83] which is designed for single-server architecture and in [84] which is designed for multi-server environment.

After a reminiscent of the work of Fujisaki and Okamoto [109], we further observed that, the treatment in the schemes [83,84,102,107] actually is equivalent to using a public-key encryption scheme that is indistinguishable against adaptive chosen cipher-text attacks (IND-CCA2). More specifically, by using a collision-resistant Hash function and a padding mechanism which is consistent with the generic padding framework in [109], the public-key encryption scheme (i.e., Elgamal) employed in [83,84,107,110] is successfully upgraded to a IND-CCA2 secure scheme in the random oracle model [111] or the real-or-random model [112] (which is a refinement of the former), achieving the accepted notion for secure encryption.

As for the scheme proposed by Son et al. in 2012 [108], it can also achieve provable security in the random oracle model and its main strategy is similar to that of [83,84,102,107], while the slight difference is that this scheme adopts the Rabin encryption algorithm and chooses the padding scheme introduced by Boneh [113]. As only one modular multiplication is performed on the user side, it is more efficient than the schemes in [83,84,102,107] where two or three exponentiations are needed. Hence, it is particularly suitable for mobile applications.

Note that security in the random oracle model does not necessarily imply that a scheme is secure in the standard model, i.e., the "real world". The limitations of the random oracle model have been well discussed in the literature (e.g., [114]). Nevertheless, it is encouraging to see that, without the cumbersome PKI, privacy-preserving two-factor authentication schemes can be built by using readily available public-key encryption schemes, while achieving high efficiency and strong notion of security under the (conditional) non-tamper-resistance assumption of the smart cards and in the random oracle model. As long as "the random oracle model constitutes a useful tool for validating natural constructions" [115], security in the random oracle model is far more reliable than in the purely heuristic approach.

In addition, though none of these five schemes [83,84,102,107,108] are designed for WSNs, they provide very strong evidence that secure and efficient two-factor authentication schemes with user anonymity for WSNs could be built by using the same cryptographic primitives. The studies in [116–119], as summarized in Table 3, also establish the feasibility and practicality of conducting a few public-key operations (e.g., RSA encryption, Elliptic Curve point multiplication and Pairing) on the resource-limited sensor nodes and smart cards.

6. Conclusion

In this work, we have focused on a hot but hard topic – privacy-preservation in WSNs. An interesting and important question that remains is, under the conditional non-tamper-resistance assumption of the smart cards, whether it is possible to construct a privacy-preserving two-factor authentication scheme for WSNs by employing only light-weight symmetric cryptographic techniques, as most of the literature has done in the past? Unfortunately, we give a negative answer to this question by empirically using two case studies and formally proving it in a widely accepted adversary model. We have also shown that this principle can be applicable to two-factor authentication schemes for universal environments. Being caught in a dilemma where public-key techniques are indispensable to achieve user anonymity but inefficient to be implemented in smart cards and sensor nodes, we have further discussed the potential solutions, and it is highly indicated that the readily-available public-key encryption schemes along with a proper padding mechanism would be promising candidates.

To the best of our knowledge, this study is the first attempt to explore the underlying rationales for preserving user privacy in two-factor authentication schemes. We believe our principle well explicates the repeated downfall in preserving user privacy in the past, and will serve as a primary guidance for the prevention of common errors in designing privacy-preserving two-factor authentication schemes in the future, providing both protocol designers and security engineers with a better understanding of the inherent cryptographic complexity in preserving user privacy. For the future work, a natural direction is to employ our proposed principle and recommended countermeasures to design privacy-preserving, secure and efficient two-factor authentication schemes for WSNs, global mobility networks, satellite networks, etc.

Acknowledgments

The authors are grateful to Prof. Chao-Hsien Chu at Pennsylvania State University for enlightening suggestions and Dr. Daojing He at South China University of Technology for insightful observations. This research was partially supported by the National Natural Science Foundation of China (NSFC) under Grants Nos. 61170263 and 61170282.

References

[1] O. Gnawali, K.-Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, E. Kohler, The tenet architecture for tiered sensor networks, in: Proc. SenSys 2006, ACM, 2006, pp. 153–166.

[2] J. Shi, R. Zhang, Y. Zhang, Secure range queries in tiered sensor networks, in: Proc. INFOCOM 2009, IEEE, 2009, pp. 945–953.

[3] D. Yang, S. Misra, X. Fang, G. Xue, J. Zhang, Two-tiered constrained relay node placement in wireless sensor networks: computational complexity and efficient approximations, IEEE Trans. Mobile Comput. 11 (8) (2012) 1399–1411.

[4] W. Zhang, H. Song, S. Zhu, G. Cao, Least privilege and privilege deprivation: towards tolerating mobile sink compromises in wireless sensor networks, in: Proc. MobiHoc 2005, ACM, 2005, pp. 378–389.

[5] D. He, J. Bu, S. Zhu, S. Chan, C. Chen, Distributed access control with privacy support in wireless sensor networks, IEEE Trans. Wireless Commun. 10 (10) (2011) 3472–3481.

[6] G.M. Yang, D.S. Wong, H.X. Wang, X.T. Deng, Two-factor mutual authentication based on smart cards and passwords, J. Comput. Syst. Sci. 74 (7) (2008) 1160–1172.

[7] D. Wang, P. Wang, Offline dictionary attack on password authentication schemes using smart cards, in: Y. Desmedt, B. Thuraisingham, K. Hamlen (Eds.), Proc. ISC 2013, LNCS, Springer-Verlag, 2014, pp. 1–16,. <http://eprint.iacr.org/2014/208.pdf>.

[8] S.J. Murdoch, S. Drimer, R. Anderson, M. Bond, Chip and PIN is broken, in: Proc. IEEE Security&Privacy 2010, IEEE Computer Society, 2010, pp. 433–446.

[9] N. Mavrogiannopoulos, A. Pashalidis, B. Preneel, Security implications in kerberos by the introduction of smart cards, in: Proc. ACM ASIACCS 2012, ACM, New York, NY, USA, 2012, pp. 59–63.

[10] J.-P. Kaps, G. Gaubatz, B. Sunar, Cryptography on a speck of dust, IEEE Comput. 40 (2) (2007) 38–44.

[11] M.L. Das, Two-factor user authentication in wireless sensor networks, IEEE Trans. Wireless Commun. 8 (3) (2009) 1086–1090.

[12] Taking the Pulse of the Planet: Epas Remote Sensing Information Gateway. <http://www.epa.gov/geoss>.

[13] The National Oceanographic Partnership Program (nopp). <http://www.nopp.org/>.

[14] B. Liu, Y. Jiang, F. Sha, R. Govindan, Cloud-enabled privacy-preserving collaborative learning for mobile sensing, in: Proc. Sensys 2012, ACM, 2012, pp. 57–70.

[15] F. Zhu, S. Carpenter, A. Kulkarni, Understanding identity exposure in pervasive computing environments, Pervasive Mobile Comput. 8 (5) (2012) 777–794.

[16] K. Mangipudi, R. Katti, A secure identification and key agreement protocol with user anonymity (SIKA), Comput. Secur. 25 (6) (2006) 420–425.

[17] J. Sun, C. Zhang, Y. Zhang, Y. Fang, Sat: a security architecture achieving anonymity and traceability in wireless mesh networks, IEEE Trans. Depend. Secur. Comput. 8 (2) (2011) 295–307.

[18] C. Lai, H. Li, X. Liang, R. Lu, K. Zhang, X. Shen, Cpal: a conditional privacy-preserving authentication with access linkability for roaming service, IEEE Internet Things J. (2014). http://dx.doi.org/10.1109/JIOT.2014.2306673.

[19] D. Hughes, V. Shmatikov, Information hiding, anonymity and privacy: a modular approach, J. Comput. Secur. 12 (1) (2004) 3–36.

[20] X. Li, W. Qiu, K. Chen, J. Li, Anonymity enhancement on robust and efficient password-authenticated key agreement using smart cards, IEEE Trans. Ind. Electron. 57 (2) (2010) 793–800.

[21] D. Wang, C. Ma, Cryptanalysis of a remote user authentication scheme with provable security for mobile client-server environment based on ECC, Inform. Fusion 14 (4) (2013) 498–503.

[22] K. Xue, P. Hong, C. Ma, A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture, J. Comput. Syst. Sci. 80 (1) (2014) 195–206.

[23] D. He, Y. Gao, S. Chan, C. Chen, J. Bu, An enhanced two-factor user authentication scheme in wireless sensor networks, Ad Hoc Sensor Wireless Netw. 10 (4) (2010) 361–371.

[24] M. Khan, K. Alghathbar, Cryptanalysis and security improvements of two-factor user authentication in wireless sensor networks, Sensors 10 (3) (2010) 2450–2459.

[25] P. Kumar, A.J. Choudhury, M. Sain, S.M. Lee, H.J. Lee, Ruasn: a robust user authentication framework for wireless sensor networks, Sensors 11 (5) (2011) 5020–5046.

[26] Q. Jiang, Z. Ma, J.F. Ma, G. Li, Security enhancement of robust user authentication framework for wireless sensor networks, China Commun. 9 (10) (2012) 103–111.

[27] K. Xue, C. Ma, P. Hong, R. Ding, A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks, J. Network Comput. Appl. 36 (1) (2013) 316–323.

[28] C. Lai, H. Li, R. Lu, X.S. Shen, SE-AKA: a secure and efficient group authentication and key agreement protocol for LTE networks, Comput. Networks 57 (17) (2013) 3492–3510.

[29] D. Nyang, M. Lee, Improvement of Das's Two-Factor Authentication Protocol in Wireless Sensor Networks, Cryptology ePrint Archive, Report 2009/631, 2009. <http://eprint.iacr.org/2009/631.pdf>.

[30] T. Chen, W. Shih, A robust mutual authentication protocol for wireless sensor networks, ETRI J. 32 (5) (2010) 704–712.

[31] H. Yeh, T. Chen, P. Liu, T. Kim, H. Wei, A secured authentication protocol for wireless sensor networks using elliptic curves cryptography, Sensors 11 (5) (2011) 4767–4779.

[32] D. Sun, J. Li, Z. Feng, Z. Cao, G. Xu, On the security and improvement of a two-factor user authentication scheme in wireless sensor networks, Pers. Ubiquitous Comput. 17 (5) (2013) 895–905.

[33] A.K. Das, P. Sharma, S. Chatterjee, J.K. Sing, A dynamic password-based user authentication scheme for hierarchical wireless sensor networks, J. Network Comput. Appl. 35 (52) (2012) 1646–1656.

[34] J.-J. Yuan, An enhanced two-factor user authentication in wireless sensor networks, Telecommun. Syst. 55 (1) (2014) 105–113.

[35] B. Vaidya, D. Makrakis, H. Mouftah, Two-factor mutual authentication with key agreement in wireless sensor networks, Secur. Commun. Netw. (2012). http://dx.doi.org/10.1002/sec.517.

[36] R. Fan, D. He, X. Pan, L. Ping, An efficient and dos-resistant user authentication scheme for two-tiered wireless sensor networks, J. Zhejinag Univ.-Sci C 12 (7) (2011) 550–560.

[37] M. Turkanovic, M. Holbl, An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks, Electron. Electr. Eng. 19 (6) (2013) 109–116.

[38] C.-T. Li, C.-Y. Weng, C.-C. Lee, C.-W. Lee, Towards secure and dynamic password based user authentication scheme in hierarchical wireless sensor networks, Int. J. Secur. Appl. 7 (3) (2013) 249–258.

[39] M. González Muñiz, P. Laud, On the (im) possibility of perennial message recognition protocols without public-key cryptography, in: Proc. ACM SAC 2011, ACM, 2011, pp. 1510–1515.

[40] D. Simon, Finding collisions on a one-way street: can secure hash functions be based on general assumptions?, in: K. Nyberg (Ed.), Proc. EUROCRYPT 1998, LNCS, vol. 1403, Springer Verlag, 1998, pp. 334–345.

[41] M. Backes, B. Pfitzmann, Limits of the cryptographic realization of dolev-yao-style xor, in: S. Vimercati, P. Syverson, D. Gollmann (Eds.), Proc. ESORICS 2005, LNCS, vol. 3679, Springer, Berlin Heidelberg, 2005, pp. 178–196.

[42] A. Buldas, A. Jrgenson, Does secure time-stamping imply collision-free hash functions?, in: W. Susilo, J. Liu, Y. Mu (Eds.), Proc. ProvSec 2007, LNCS, vol. 4784, Springer, Berlin Heidelberg, 2007, pp. 138–150.

[43] R. Impagliazzo, S. Rudich, Limits on the provable consequences of one-way permutations, in: Proc. 21th Ann. ACM Symp. on Theory of Computing (STOC 1989), ACM, 1989, pp. 44–61.

[44] S. Halevi, H. Krawczyk, Public-key cryptography and password protocols, ACM Trans. Inf. Syst. Secur. 2 (3) (1999) 230–268.

[45] D. Park, C. Boyd, S.-J. Moon, Forward secrecy and its application to future mobile communications security, in: H. Imai, Y. Zheng (Eds.), Proc. PKC 2000, LNCS, vol. 1751, Springer, Berlin/ Heidelberg, 2000, pp. 433–445.

[46] M.-H. Nguyen, The relationship between password-authenticated key exchange and other cryptographic primitives, in: J. Kilian (Ed.), Proc. TCC 2006, LNCS, vol. 3378, Springer Verlag, 2005, pp. 457–475.

[47] P. Zeng, Z. Cao, K.-k. Choo, S. Wang, On the anonymity of some authentication schemes for wireless communications, IEEE Commun. Lett. 13 (3) (2009) 170–171.

[48] J. Zhu, J. Ma, A new authentication scheme with anonymity for wireless environments, IEEE Trans. Consum. Electron. 50 (1) (2004) 231–235.

[49] C.-C. Lee, M.-S. Hwang, I.-E. Liao, Security enhancement on a new authentication scheme with anonymity for wireless environments, IEEE Trans. Ind. Electron. 53 (5) (2006) 1683–1687.

[50] C.-C. Wu, W.-B. Lee, W.-J. Tsaur, A secure authentication scheme with anonymity for wireless communications, IEEE Commun. Lett. 12 (10) (2008) 722–723.

[51] C.-T. Li, C.-Y. Weng, C.-C. Lee, An advanced temporal credential-based security scheme with mutual authentication and key agreement for WSNs, Sensors 13 (8) (2013) 9589–9603.
[52] P. Kumar, S.-G. Lee, H.-J. Lee, E-sap: Efficient-strong authentication protocol for healthcare applications using wireless medical sensor networks, Sensors 12 (2) (2012) 1625–1647.
[53] T.S. Messerges, E.A. Dabbish, R.H. Sloan, Examining smart-card security under the threat of power analysis attacks, IEEE Trans. Comput. 51 (5) (2002) 541–552.
[54] K. Markantonakis, M. Tunstall, G. Hancke, I. Askoxylakis, K. Mayes, Attacking smart card systems: theory and practice, Inform. Secur. Tech. Rep. 14 (2) (2009) 46–56.
[55] T.H. Kim, C. Kim, I. Park, Side channel analysis attacks using am demodulation on commercial smart cards with seed, J. Syst. Soft. 85 (12) (2012) 2899–2908.
[56] J. Lancia, Java card combined attacks with localization-agnostic fault injection, in: S. Mangard (Ed.), Proc. of the 11th Int. Conf. on Smart Card Research and Advanced Applications, LNCS, vol. 7771, Springer, Verlag, 2013, pp. 31–45.
[57] K. Nohl, D. Evans, S. Starbug, H. Plötz, Reverse-engineering a cryptographic rfid tag, in: Proc. 17th USENIX Security Symp. (USENIX Security 2008), USENIX Association, 2008, pp. 185–193.
[58] J.-S. Leu, W.-B. Hsieh, Efficient and secure dynamic id-based remote user authentication scheme for distributed systems using smart cards, IET Inf. Secur. 8 (2) (2014) 104–113.
[59] H. Li, Y. Yang, L. Pang, An efficient authentication protocol with user anonymity for mobile networks, in: Proc. WCNC 2013, IEEE, 2013, pp. 1842–1847.
[60] D. He, M. Ma, Y. Zhang, C. Chen, J. Bu, A strong user authentication scheme with smart cards for wireless communications, Comput. Commun. 34 (3) (2011) 367–374.
[61] C. Chen, D. He, S. Chan, J. Bu, R. Fan, Lightweight and provably secure user authentication with anonymity for the global mobility network, Int. J. Commun. Syst. 24 (3) (2011) 347–362.
[62] X. Li, Y. Xiong, J. Ma, W. Wang, An enhanced and security dynamic identity based authentication protocol for multi-server architecture using smart cards, J. Network Comput. Appl. 35 (2) (2012) 763–769.
[63] F. Wen, X. Li, An improved dynamic id-based remote user authentication with key agreement scheme, Comput. Electri. Eng. 38 (2) (2012) 381–387.
[64] M. Khan, S. Kim, Cryptanalysis and security enhancement of a more efficient & secure dynamic id-based remote user authentication scheme', Comput. Commun. 34 (3) (2011) 305–309.
[65] M. Turkanović, B. Brumen, M. 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.
[66] C.-C. Chang, T.-F. Cheng, H.-L. Wu, An authentication and key agreement protocol for satellite communications, Int. J. Commun. Syst. (2013). http://dx.doi.org/10.1002/dac.2448.
[67] S. Shin, H. Yeh, K. Kim, An efficient secure authentication scheme with user anonymity for roaming user in ubiquitous networks, Peer-to-Peer Network Appl. (2013). http://dx.doi.org/10.1007/s12083-013-0218-2.
[68] F. Bao, R. Deng, Privacy protection for transactions of digital goods, in: S. Qing, T. Okamoto, J. Zhou (Eds.), Proc. ICICS 2001, LNCS, vol. 2229, Springer-Verlag, 2001, pp. 202–213.
[69] M.L. Das, A. Saxena, V.P. Gulati, A dynamic id-based remote user authentication scheme, IEEE Trans. Consum. Electron. 50 (2) (2004) 629–631.
[70] Y.-P. Liao, S.-S. Wang, A secure dynamic id based remote user authentication scheme for multi-server environment, Comput. Stand. & Inter. 31 (1) (2009) 24–29.
[71] D. Wang, P. Wang, Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks, Ad Hoc Networks 20 (2014) 1–15.
[72] Y.G. Wang, Password protected smart card and memory stick authentication against off-line dictionary attacks, Proc. SEC 2012, IFIP AICT, vol. 376, Springer, Boston, 2012, pp. 489–500.
[73] S. Zhong, F. Wu, A collusion-resistant routing scheme for noncooperative wireless ad hoc networks, IEEE/ACM Trans. Network 18 (2) (2010) 582–595.
[74] Q. Xiao, K. Bu, Z. Wang, B. Xiao, Robust localization against outliers in wireless sensor networks, ACM Trans. Sensor Network 9 (2) (2013) 24.
[75] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inform. Theory 29 (2) (1983) 198–208.
[76] Miracl Library, Shamus Software Ltd. <http://www.shamus.ie/index.php?page=home>.
[77] J. Bonneau, M. Just, G. Matthews, Whats in a name? in: R. Sion (Ed.), Proc. FC 2010, LNCS, vol. 6052, Springer, 2010, pp. 98–113.
[78] J. Bonneau, The science of guessing: analyzing an anonymized corpus of 70 million passwords, in: Proc. IEEE S&P 2012, IEEE Computer Society, 2012, pp. 538–552.
[79] M. Dell'Amico, P. Michiardi, Y. Roudier, Password strength: An empirical analysis, in: Proc. INFOCOM 2010, 2010, pp. 1–9.
[80] H.-M. Sun, W.-C. Ting, K.-H. Wang, On the security of chien's ultralightweight RFID authentication protocol, IEEE Trans. Depend. Secur. Comput. 8 (2) (2011) 315–317.
[81] H. Wang, Y. Zhang, On the security of a ticket-based anonymity system with traceability property in wireless mesh networks, IEEE Trans. Depend. Secur. Comput. 9 (3) (2012) 443–446.
[82] G. Wang, J. Yu, Q. Xie, Security analysis of a single sign-on mechanism for distributed computer networks, IEEE Trans. Ind. Inform. 9 (1) (2013) 294–302.
[83] D. Wang, P. Wang, C.G. Ma, Z. Chen, iPass: robust smart card based password authentication scheme against smart card loss problem, J. Comput. Syst. Sci., (2014) (in press). <http://eprint.iacr.org/2012/439.pdf>.
[84] M. Khan, D. He, A new dynamic identity-based authentication protocol for multi-server environment using elliptic curve cryptography, Secur. Commun. Netw. 5 (11) (2012) 1260–1266.
[85] C. Ma, D. Wang, S.D. Zhao, Security flaws in two improved remote user authentication schemes using smart cards, Int. J. Commun. Syst. (2012). http://dx.doi.org/10.1002/dac.2468.
[86] D. Wang, P. Wang, J. Liu, Improved privacy-preserving authentication scheme for roaming service in mobile networks, in: Proc. WCNC 2014, 2014, pp. 3178–3183.
[87] D. Wang, C.G. Ma, D.L. Gu, Z.S. Cui, Cryptanalysis of two dynamic id-based remote user authentication schemes for multi-server architecture, in: L. Xu, Y. Mu (Eds.), Proc. NSS 2012, LNCS, vol. 7645, Springer, Berlin/Heidelberg, 2012, pp. 462–475.
[88] P. Kumar, A. Gurtov, M. Ylianttila, S.-G. Lee, H. Lee, A strong authentication scheme with user privacy for wireless sensor networks., ETRI J. 35 (5) (2013) 889–899.
[89] F. Wu, L. Xu, Security analysis and improvement of a privacy authentication scheme for telecare medical information systems, J. Med. Syst. (2014). http://dx.doi.org/10.1007/s10916-013-9958-z.
[90] M.K. Khan, S. Kumari, Cryptanalysis and improvement of an efficient and secure dynamic id-based authentication scheme for telecare medical information systems, Secur. Commun. Networks (2013). http://dx.doi.org/10.1002/sec.791.
[91] M.-C. Chuang, J.-F. Lee, M.-C. Chen, Spam: A secure password authentication mechanism for seamless handover in proxy mobile ipv6 networks, IEEE Syst. J. 7 (1) (2013) 102–113.
[92] J. Katz, R. Ostrovsky, M. Yung, Efficient and secure authenticated key exchange using weak passwords, J. ACM 57 (1) (2009) 1–39.
[93] D. Wang, C.G. Ma, On the Security of Some Smart-Card-Based Remote User Authentication Schemes for wsn, Cryptology ePrint Archive, Report 2012/581, 2012. <http://eprint.iacr.org/2012/581.pdf>.
[94] F. Wen, W. Susilo, G. Yang, A secure and effective anonymous user authentication scheme for roaming service in global mobility networks, Wireless Pers. Commun. (2013). http://dx.doi.org/10.1007/s11277-013-1243-4.
[95] H.-C. Hsiang, W.-K. Shih, Improvement of the secure dynamic id based remote user authentication scheme for multi-server environment, Comput. Stand. Interfaces 31 (6) (2009) 1118–1123.
[96] M. Kang, S. Rhee, Y. Choi, Improved user authentication scheme with user anonymity for wireless communications, IEICE Trans. Fund. Electron. Commun. Comput. Sci. 94 (2) (2011) 860–864.
[97] X. Li, J. Ma, W. Wang, J. Zhang, A novel smart card and dynamic id based remote user authentication scheme for multi-server environments, Math. Comput. Model. 58 (2) (2013) 85–95.
[98] J. Kim, D. Lee, W. Jeon, Y. Lee, D. Won, Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks, Sensors 14 (4) (2014) 6443–6462.
[99] S. Kumari, M.K. Khan, X. Li, An improved remote user authentication scheme with key agreement, Comput. Electrical Eng. (2014). http://dx.doi.org/10.1016/j.compeleceng.2014.05.007.
[100] X. Leroy, Smart card security from a programming language and static analysis perspective, INRIA Rocquencourt & Trusted Logic, Tecnical Report, 2013. <http://pauillac.inria.fr/xleroy/talks/language-security-etaps03.pdf>.
[101] D. Sun, J. Huai, J. Sun, J. Li, J. Zhang, Z. Feng, Improvements of Juang et al.'s password-authenticated key agreement scheme using smart cards, IEEE Trans. Ind. Electron. 56 (6) (2009) 2284–2291.

[102] T. Zhou, J. Xu, Provable secure authentication protocol with anonymity for roaming service in global mobility networks, Comput. Network 55 (1) (2011) 205–213.
[103] D. He, C. Chen, J. Bu, S. Chan, Y. Zhang, Security and efficiency in roaming services for wireless networks: challenges, approaches, and prospects, IEEE Commun. Mag. 51 (2) (2013) 142–150.
[104] D. He, C. Chen, S. Chan, J. Bu, Secure and efficient handover authentication based on bilinear pairing functions, IEEE Trans. Wireless Commun. 11 (1) (2012) 48–53.
[105] D. Chaum, E. Heyst, Group signatures, in: D. Davies (Ed.), Proc. EUROCRYPT 1991, LNCS, vol. 547, Springer Verlag, 1991, pp. 257–265.
[106] J. Ren, L. Harn, An efficient threshold anonymous authentication scheme for privacy-preserving communications, IEEE Trans. Wireless Commun. 12 (3) (2013) 1018–1025.
[107] J. Xu, W.-T. Zhu, A generic framework for anonymous authentication in mobile networks, J. Comput. Sci. Technol. 28 (4) (2013) 732–742.
[108] K. Son, D. Han, D. Won, A privacy-protecting authentication scheme for roaming services with smart cards, IEICE Trans. Commun. 95 (5) (2012) 1819–1821.
[109] E. Fujisaki, T. Okamoto, Secure integration of asymmetric and symmetric encryption schemes, in: M. Wiener (Ed.), Proc. CRYPTO 1999, LNCS, vol. 1666, Springer, Verlag, 1999, pp. 537–554.
[110] T. Zhou, J. Xu, Provable secure authentication protocol with anonymity for roaming service in global mobility networks, Comput. Netw. 55 (1) (2011) 205–213.
[111] M. Bellare, D. Pointcheval, P. Rogaway, Authenticated key exchange secure against dictionary attacks, Proc. EUROCRYPT 2000, LNCS, vol. 1807, Springer Verlag, 2000, pp. 139–155.
[112] M. Abdalla, P.-A. Fouque, D. Pointcheval, Password-based authenticated key exchange in the three-party setting, in: S. Vaudenay (Ed.), Proc. PKC 2005, LNCS, vol. 3386, Springer Verlag, 2005, pp. 65–84.
[113] D. Boneh, Simplified OAEP for the RSA and Rabin functions, in: J. Kilian (Ed.), Proc. CRYPTO 2001, LNCS, vol. 2139, Springer Verlag, 2001, pp. 275–291.
[114] R. Canetti, O. Goldreich, S. Halevi, The random oracle methodology, revisited, J. ACM 51 (4) (2004) 557–594.
[115] D. Boneh, M. Franklin, Identity-based encryption from the weil pairing, SIAM J. Comput. 32 (3) (2003) 586–615.
[116] S.-Y. Chang, Y.-H. Lin, H.-M. Sun, M.-E. Wu, Practical rsa signature scheme based on periodical rekeying for wireless sensor networks, ACM Trans. Sen. Network 8 (2) (2012) 13.
[117] A.S. Wander, N. Gura, H. Eberle, V. Gupta, S.C. Shantz, Energy analysis of public-key cryptography for wireless sensor networks, in: Proc. PerCom 2005, IEEE, 2005, pp. 324–328.
[118] M. Scott, N. Costigan, W. Abdulwahab, Implementing cryptographic pairings on smartcards, in: L. Goubin, M. Matsui (Eds.), CHES 2006, LNCS, vol. 4249, Springer Verlag, 2006, pp. 134–147.
[119] P. Szczechowiak, A. Kargl, M. Scott, M. Collier, On the application of pairing based cryptography to wireless sensor networks, in: Proc. WiSec 2009, ACM, 2009, pp. 1–12.

Ding Wang received his B.S. Degree in Information Security from Nankai University, Tianjin, China, in 2008. Then he went to Information Engineering University (Zhengzhou) to work toward Information Security Engineering. Now he is pursuing his Ph.D. degree at Peking University, Beijing, China. He has published more than 20 refereed research papers at Elsevier, IEEE and Wiley journals, and conferences such as DBSec 2012, ICICS 2012, ISC 2013, SecureComm 2014 and WCNC 2014. He was awarded the Top-Ten Distinguished Graduate Academic Star of the University in 2012. His research interests include cryptography and wireless network security.

Ping Wang received his B.S. degree from University of Electronic Science and Technology of China in 1983, and Ph.D. degree in Computer Science from University of Massachusetts, USA, in 1996. He has served as chief engineer in Open Software Foundation and Lucent Technologies Ltd. Currently, he is a full-time Professor and director of the PKU-PSU Joint Laboratory of Intelligent Computing and Smart Sensing. He served as technique committee co-chairs of RFIDSec'11 Asia. He has wide interests in system security, system software and distributed computing.