This paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE WCNC 2014 proceedings
This paper is to be presented as part of the 2014 IEEE Wireless Communications and Networking Conference: Track 3 – Mobile and Wireless Networks

Improved Privacy-Preserving Authentication Scheme for Roaming Service in Mobile Networks

Ding Wang†*, Ping Wang†*, Jing Liu†
† School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
*Key laboratory of High Confidence Software Technologies (Peking University), Ministry of Education, China
Email: wangdingg@mail.nankai.edu.cn; pwang@pku.edu.cn; liujing@ss.pku.edu.cn

***Abstract*—User authentication is an important security mechanism that allows mobile users to be granted access to roaming service offered by the foreign agent with assistance of the home agent in mobile networks. While security-related issues have been well studied, how to preserve user privacy in this type of protocols still remains an open problem. In this paper, we revisit the privacy-preserving two-factor authentication scheme presented by Li *et al.* at WCNC 2013. We show that, despite being armed with a formal security proof, this scheme actually cannot achieve user anonymity and is insecure against offline password guessing attacks, and thus, it is not recommended for practical applications. Then, we figure out how to fix these identified drawbacks, and suggest an enhanced scheme with better security and reasonable efficiency. Further, we conjecture that under the non-tamper-resistant assumption of the smart cards, only symmetric-key techniques are intrinsically insufficient to attain user anonymity.**

***keywords*–Mobile networks; Roaming service; Password authentication; Smart card; User anonymity.**

I. Introduction

With the rapid development of wireless communications technologies, such as AMPS, GSM, GPRS, 3G and 4G, over the last couple of decades, it is becoming more and more convenient for users to enjoy desired resources from distributed service providers by using mobile devices (e.g., PDAs, PMPs, Smart phones and Laptops) at any time and anywhere [1], [2]. Due to its dynamic topology and broadcast communication nature, wireless networks are much more difficult to achieve the same level of security with that of wired networks. To further complicate matters, new concerns such as user privacy and seamless access arise at an ever increasing pace.

In mobile networks, to provide privacy-preserving and secure roaming service for a mobile user between her home network and a foreign network (see Fig.1), smart-card-based password authentication [3], [4], or the so-called two-factor authentication [5], has gained considerable interest because of its simplicity, portability and cryptographic capacity. In such authentication schemes, each mobile user MU is equipped with a smart card (e.g., SIM card) and a low-entropy password (e.g., a four or six digit PIN). Whenever MU roams to a foreign network managed by a foreign agent FA, she performs two-factor authentication with FA before being granted network access, with the help of her home agent HA in her home network. Only the user who can present the smart card and the corresponding password can pass the verification of FA and HA. Upon a successful run of the scheme, MN and FA establish a session key that might be used to protect their ensuing data communications.

In 2004, Zhu and Ma [6] proposed the first two-factor authentication scheme for roaming service that is claimed to preserve user anonymity with respect to the adversaries as well as the (curious) foreign server. Before we proceed, it is worth noting that in the context of user authentication, the notion of user anonymity mainly has two properties [3], [7]: 1) a basic property, i.e. user identity-protection, which guarantees that an attacker cannot determine the target user's identity from the protocol transcripts; and 2) an advanced property, i.e. user un-traceability, which ensures that an attacker can neither figure out who the user is nor discern whether two sessions are generated by the same user. Obviously, a scheme that only achieves the basic anonymity property, i.e. user identity-protection, cannot prevent the attacker from tracing user activities. This is especially undesirable for mobile networks, and hence Zhu-Ma's scheme as well as all the other schemes mentioned in this paper strive to achieve user un-traceability. [1]

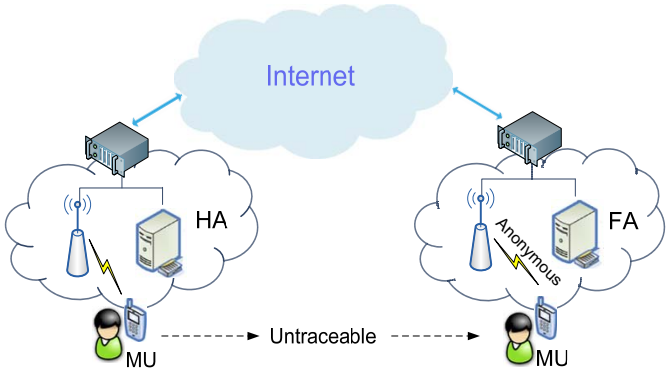

Fig. 1. Privacy-reserving user authentication for roaming service

Shortly after Zhu-Ma's scheme was presented, Lee *et al.* [8] showed that it fails to support perfect backward secrecy and mutual authentication, and proposed an enhanced version. Unfortunately, in 2008 Wu [9] demonstrated that both the schemes of Zhu-Ma and Lee *et al.* actually cannot provide user anonymity, and he further suggested a simple fix. Later, Zeng *et al.* [10] revealed that, due to a structural mistake made in Zhu-Ma's scheme, this scheme [6] and its two successors [8], [9] are unlikely to attain user anonymity. In 2009, Chang *et al.* [11] reported that Lee *et al.*'s enhanced protocol still suffers from the impersonation attack and proposed yet another improved version so as to overcome the identified deficiencies.

In 2011, Zhou and Xu [12] pointed out that previous schemes (e.g., [6], [8], [9], [11]) all fail to preserve user anonymity and suffer from various other defects. Accordingly, they presented

[1] Hereafter, unless otherwise specified, by "user anonymity" we will always mean "user un-traceability".

978-1-4799-3083-8/14/$31.00 ©2014 IEEE

a new scheme which is claimed to be immune to all known attacks. In the meantime, Xu *et al.* [13] also put forward a scheme to meet the stringent security and privacy requirements of user authentication for roaming service.

Unlike prior schemes (e.g., [6], [8], [9], [11]) that only employ lightweight symmetric-key primitives to provide user anonymity, the schemes in [12], [13] need to perform two or three modular exponentiations on the mobile user side. As such asymmetric-key operations are comparatively expensive, it is undesirable for mobile networks where users usually hold low-power devices. Consequently, at WCNC 2013, Li *et al.* [14] stepped back to the old direction and endeavored to only employ symmetric-key techniques to design a secure and privacy-preserving two-factor scheme for roaming service.

Contributions. In this paper, however, we demonstrate that although Li *et al.*'s scheme [14] is efficient and has been equipped with a formal proof, it fails to meet its essential goal – user anonymity. Besides, it is not a truly two-factor scheme either, for it is vulnerable to an offline password guessing attack in which an attacker can obtain the password factor once the smart-card factor is compromised. *Now a paradox arises*: How can a protocol that was formally proven secure later be found insecure? We provide an answer to this question by showing that its security proof is flawed. Furthermore, an effective fix is suggested to tackle the identified problems, making up the missing security and privacy provisions while still maintaining acceptable efficiency.

Organization. The paper is organized as follows: in Section II, we review Li *et al.*'s scheme. Section III describes the weaknesses of Li *et al.*'s scheme. Our improved scheme is presented and evaluated in Section IV and Section V, respectively. Section VI concludes the paper and points out future directions.

II. Review of Li *et al.*'s scheme

For a self-contained discussion, in this section we briefly review Li *et al.*'s scheme [14]. For ease of presentation, we employ some intuitive notations as listed in Table 1.

A. Registration phase

In this phase, whenever a mobile user MU enrolls in her home agent HA, the following steps are performed:

(1) MU chooses her identity ID_{MU}, password PW_{MU}.
(2) $MU \Rightarrow HA : \{ID_{MU}, PW_{MU}\}$.
(3) HA initializes C_{new} and C_{old} to 0 and sets t_{last} to the current timestamp. Then HA stores $\{C_{new}$, C_{old}, ID_{MU}, $t_{last}\}$ and $\{ID_{HA}, x\}$ separately.
(4) $HA \Rightarrow MU : \{h(x), ID_{HA}\}$.
(5) MU sets $C = 0$, computes $u = h(ID_{MU} \parallel ID_{HA}) \oplus PW_{MU}$ and stores $\{C$, ID_{MU}, ID_{HA}, u, $h(x)\}$ in the smart card.

B. Login phase

When MU roams into a foreign network managed by a foreign agent FA, she authenticates herself to FA to show that she is a legitimate subscriber of her home agent HA. This phase involves the following steps:

(1) MU inserts her smart card into a card reader and inputs her password PW_{MU}.
(2) The smart card computes $D = u \oplus PW_{MU}$ and checks whether $D \stackrel{?}{=} h(ID_{MU} \parallel ID_{HA})$. If the equality does not hold, it rejects the login request.
(3) The smart card generates a random number x_0, reads current timestamp t_{MU}, and computes $SID = h(x) \oplus h(ID_{MU} \parallel x_0)$ and $V_1 = h(SID \parallel C \parallel h(x) \parallel t_{MU})$.
(4) $MU \rightarrow FA : m_1 = \{SID, V_1, C, ID_{HA}, x_0, t_{MU}\}$.

C. Mutual authentication phase

(1) Upon receiving the login request m_1, FA generates a random number y_0, computes $H_{mac} = h(y_0 \parallel ID_{FA} \parallel K_{FH} \parallel V_1)$, where K_{FH} is a pre-shared symmetric key between FA and HA.
(2) $FA \rightarrow HA : m_2 = \{m_1, y_0, H_{mac}, ID_{FA}\}$.
(3) Upon receiving m_2, HA performs the following steps:
   a) HA retrieves K_{FH} according to ID_{FA} and checks $H_{mac} \stackrel{?}{=} h(y_0 \parallel ID_{FA} \parallel K_{FH} \parallel V_1)$. If they are not equal, HA terminates.
   b) HA checks whether $V_1 \stackrel{?}{=} h(SID \parallel C \parallel h(x) \parallel t_{MU})$. If they are equal, HA proceeds to check whether t_{MU} is larger than t_{last}.
   c) If $C = 0$, HA searches ID_{MU} that satisfies: 1) $SID \oplus h(x) == h(ID_{MU} \parallel x_0)$; 2) the corresponding C_{old} or C_{new} is 0. If there exists such an ID_{MU} , then $z = old$ or new, and it proceeds to the next step. If $C \neq 0$ and C_{old} or C_{new} is equal to C, it sets $z = old$ or new; otherwise, it terminates.
   d) If $z = new$, then $C_{old} = C_{new}$, $C_{new} = PRNG(x_0 \parallel h(ID_{MU} \parallel ID_{HA}))$; if $z = old$, then $C_{new} = PRNG(x_0 \parallel h(ID_{MU} \parallel ID_{HA}))$. HA computes $V_2 = h(h(ID_{MU} \parallel ID_{HA}) \parallel ID_{FA} \parallel x_0 \parallel C)$, $E = K_{FH} \oplus h(ID_{MU} \parallel ID_{HA}) \oplus h(x)$, $S_2 = h(V_2 \oplus E \oplus K_{FH} \oplus y_0)$ and $t_{last} = t_{MU}$.
(4) $HA \rightarrow FA : m_3 = \{S_2, V_2, E\}$.
(5) After FA receives message m_3 , it verifies $S_2 \stackrel{?}{=} h(V_2 \oplus E \oplus K_{FH} \oplus y_0)$. If they are not equal, FA terminates. Otherwise, FA computes the session key $sk_{FA} = h(E \oplus K_{FH} \oplus x_0 \oplus y_0)$ and generates a temporary digital certificate $TCertu$ for MU.
(6) $FA \rightarrow MU : m_4 = \{y_0, E_{sk_{FA}}(TCertu \parallel V_2 \parallel ID_{FA})\}$.
(7) On receiving message m_4, MU computes the session key $sk_{MU} = h(D \oplus h(x) \oplus x_0 \oplus y_0)$, decrypts $E_{sk_{FA}}(TCertu \parallel V_2 \parallel ID_{FA})$ with sk_{MU} and checks whether V_2 is equal to $h(D \parallel ID_{FA} \parallel x_0 \parallel C)$. If they are equal, the mobile user MU updates her index $C = PRNG(x_0 \parallel D)$; otherwise, MU rejects.

D. Session key update phase and password change phase

To enhance the efficiency and security, while MU stays with the same FA, the i^{th} session key sk^i can be derived from the unexpired previous secret knowledge and thus the number of messages exchanged is kept as few as possible. To enable periodical password change, password change phase is provided. We skip the review of these two phases as they have little relevance regarding our discussions.

Remark 1. There are some obscure mechanisms in Li *et al.*'s scheme [14] which tend to be quite incomprehensible and

TABLE I
NOTATIONS AND ABBREVIATIONS

Symbol	Description	Symbol	Description
MU	mobile user	x	long-term symmetric key of HA
HA	home agent	K_{FH}	a pre-shared symmetric key between FA and HA
FA	foreign agent	p, q, n	p and q are two larger prime numbers, and $n = pq$
$\mathcal{A}$	the adversary	e, d	e is a prime number and d is an integer, where $ed = 1 \bmod (p-1)(q-1)$
ID_{MU}	identity of mobile user MU	$\|$	the string concatenation operation
PW_{MU}	password of mobile user MU	$\oplus$	the bitwise XOR operation
$\Rightarrow$	a secure channel	$E/D(\cdot)$	symmetric encryption/decryption algorithm
$\rightarrow$	a common channel	$h(\cdot)$	collision free one-way hash function

even unreasonable. For example, in Step 2 of the registration phase, "HA initializes C_{new} and C_{old} to 0"; in Step 3 of the authentication phase, "If $z = new$, then HA sets $C_{old} = C_{new}$, $C_{new} = PRNG(x_0 \parallel h(ID_{MU} \parallel ID_{HA}))$; if $z = old$, then HA sets $C_{new} = PRNG(x_0 \parallel h(ID_{MU} \parallel ID_{HA}))$". It is unclear what on earth these operations are for. Moreover, the variables C, C_{new}, C_{old} and t_{last} only appear in the definition of Li *et al.*'s scheme, and they have never been mentioned in their later security analysis and performance evaluation. It seems that these variables are introduced just to complicate the protocol, incurring increased computation overhead and unnecessary communication cost. This obscurity does not affect our following cryptanalysis however.

III. CRYPTANALYSIS OF LI *et al.*'S SCHEME

Although Li *et al.*'s scheme [14] possesses many desirable features such as local password update, fast reconnect and high efficiency, it is completely insecure in the presence of an active adversary. To show this, we present two practical attacks on it.

A. User anonymity violation attack

As with most related literature (e.g., [4], [5], [12], [13]), the following two assumptions are explicitly made about the capabilities of the adversary when Li *et al.* [14] analyzed the security of their scheme: (1) $\mathcal{A}$ has total control over the communication channel among the user MU, the foreign agent FA and the home agent HA, which is consistent with the common adversary model for distributed computing, called the Dolev-Yao model; (2) $\mathcal{A}$ can break either *the password factor* of the user by using a malicious card reader *or the smart-card factor* of the user by side-channel attacks [15], [16], but not both to avoid trivial case. With these two assumptions, the adversary $\mathcal{A}$, who is also a legitimate user of HA, can successfully breach the claimed goal of user anonymity:

Step 1. Extracts $h(x)$ from $\mathcal{A}$'s own smart card, note that $h(x)$ is shared between HA and all its subscribers.
Step 2. Intercepts a login request message, say $m_1 = \{SID, V_1, C, ID_{HA}, x_0, t_{MU}\}$, sent by MU;
Step 3. Guesses the value of ID_{MU} to be ID^*_{MU} from a dictionary space $\mathcal{D}_{id}$.
Step 4. Computes $SID^* = h(ID^*_{MU} \parallel x_0) \oplus h(x)$, where x_0 is intercepted in Step 2.
Step 5. Verifies the correctness of ID^*_{MU} by checking if the computed SID^* is equal to the intercepted SID.
Step 6. Repeats the above Steps $3 \sim 5$ until the correct value of ID_{MU} is found.

To lunch the above attack, $\mathcal{A}$ only needs to reveal the secret $h(x)$ from her own smart card and to eavesdrop a single run of the protocol, then she can offline guess the victim's identity without interacting with FA or HA. Note that, revealing data from $\mathcal{A}$'s own smart card is much easier than from MU's (lost) smart card. In this regard, our attack is quite realistic.

Let $|\mathcal{D}_{id}|$ denote the number of identities in $\mathcal{D}_{id}$. The running time of our attack is $\mathcal{O}(|\mathcal{D}_{id}| * (T_H + T_{XOR}))$, where T_H and T_{XOR} are the computation time for Hash operation and XOR operation, respectively. As $|\mathcal{D}_{id}|$ is very limited in practice, e.g. $|\mathcal{D}_{id}| \leq |\mathcal{D}_{pw}| \leq 10^6$ [17], [18], this attack can be completed in seconds on a Laptop.

B. Offline password guessing attack

Offline password guessing attack is the most damaging threat that a sound password-based protocol should be able to thwart [19]. In this attack, the adversary records past communication messages, and then *offline* search through the password dictionary and look for a password that is consistent with the recorded communication. As the password dictionary often is limited, it is highly likely that the adversary will succeed.

Now let's see how this attack could be successfully launched with Li *et al.*'s scheme in place. In case a legitimate user MU's smart card is somehow obtained (stolen or accidently picked up) by $\mathcal{A}$, and the stored secret u can be revealed by some means (e.g., side-channel attacks [15], [16]). With the extracted u, $\mathcal{A}$ can obtain MU's password PW_{MU} as follows:

Step 1. Computes MU's identity ID_{MU} as described in the above section.
Step 2. Guesses MU's password to be PW^*_{MU} from dictionary space $\mathcal{D}_{pw}$.
Step 3. Computes $u^* = h(ID_{MU} \| ID_{HA}) \oplus PW^*_{MU}$. Note that since $\mathcal{A}$ is a legitimate (but malicious) user, she knows ID_{HA}.
Step 4. Verifies the correctness of PW^*_{MU} by checking if the computed u^* is equal to the extracted u.
Step 5. Repeats the above Steps $2 \sim 4$ until the correct value of PW_{MU} is found.

This attack reveals that, Li *et al.*'s protocol [14] is actually not a two-factor scheme, for a compromise of a single factor (i.e., the smart-card factor) will lead to the compromise of the remaining factor (i.e., the password factor).

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 cryptographic library MIRACL [20], a multi-precision and rational arithmetic C/C++ library. We carry out experiments on Laptop PCs with different computation power. For accuracy, each operation is repeated for one thousand times. Table II lists the experimental data for related operations on common Laptops.

TABLE II
COMPUTATION EVALUATION (IN MICROSECONDS) OF RELATED OPERATIONS ON COMMON LAPTOP PCS

Experimental platform (common PCs)	Hash operation T_H (SHA-1)	Other lightweight operations (e.g., XOR and Concatenation)
Intel T5870 2.00 GHz	2.437 µs	0.011 µs
Intel i5750 2.66 GHz	1.980 µs	0.009 µs
Pentium IV 3.06 GHz	1.526 µs	0.008 µs

Let $|\mathcal{D}_{pw}|$ denote the number of passwords in $\mathcal{D}_{pw}$. The running time of the above attack procedure is $\mathcal{O}((T_H+T_{XOR})*(|\mathcal{D}_{id}| + |\mathcal{D}_{pw}|))$. Consequently, the time for $\mathcal{A}$ to recover MU's password is a linear function of the sum of $|\mathcal{D}_{id}|$ and $|\mathcal{D}_{pw}|$. As $|\mathcal{D}_{id}|$ and $|\mathcal{D}_{pw}|$ are very limited in practice, e.g. $|\mathcal{D}_{id}| \leq |\mathcal{D}_{pw}| \leq 10^6$ [17], [18], this attack can be completed in 5($\approx$ (2.437 µs+0.011 µs)$*2*10^6$) seconds on a common Laptop.

Remark 2. Privacy-preserving two-factor authentication schemes have attracted considerable interest from the research community, yet relatively little rationale has been uncovered. We have reviewed more than one hundred recently proposed schemes (some of our recent cryptanalysis results include [7], [21]), and observed that all these two-factor schemes that do not involve asymmetric-key operations on the user side are invariably unable to preserve user anonymity. Quite recent failures include [4], [22], [23]. We conjecture that the strategy of only using symmetric-key techniques to achieve user anonymity is *intrinsically infeasible*, and promote the formal proof of this principle as one open problem.

IV. PROOF ANALYSIS AND SECURITY ENHANCEMENT

Generally speaking, a password-based protocol achieving semantic security or the so-called "AKE security" [19] can provide an acceptable level of security such as resistance to offline password guessing attack, impersonation attack and replay attack. Li *et al.*'s protocol [14] carries a claimed proof of its semantic security, yet as we have shown, it is prone to offline password guessing attack once the smart-card factor is broken.

Now a paradoxical question arises: *How can a protocol that was proved secure later was found to be insecure?* To answer this question, we investigate into the formal security proof of Li *et al.*'s protocol. After figuring out the fundamental flaw in their reasoning of the proof, we suggest an enhancement to overcome the identified security defects.

A. Flaw in Li et al.'s security proof

The security model adopted by Li *et al.* is based on the work of Zhou-Xu [12], while the latter is essentially a variant of the random oracle model introduced by Bellare *et al.* [19]. The common basic idea that lies behind a proof in the random oracle model is: 1) Taking any hash function as an oracle which outputs a random value for each new query; 2) Exploiting $\mathcal{A}$ to construct adversaries for each of the underlying cryptographic primitives (e.g., computational Diffie-Hellman assumption and large integer factorization assumption) in such a way that if $\mathcal{A}$ manages to break protocol $\mathcal{P}$, then at least one of these adversaries manages to solve an underlying primitive. Since these primitives are widely considered computationally hard problems, no adversary can succeed in breaking any one of them in polynomial time and hence the protocol remains secure.

The tactic employed by Li *et al.* to prove semantic security of the session key is similar to that of [24], in which the goal is accomplished through a series of hybrid experiments Exp_n ($n = 0, 1, 2, 3$), starting with the real attack Exp_0 and ending in an experiment Exp_3 where $\mathcal{A}$'s advantage is 0, and for which we can bound the difference in $\mathcal{A}$'s advantage between any two consecutive experiments. This yields a bound on the adversary's advantage when attacking the original protocol $\mathcal{P}$. Though a sequence of experiments are incrementally defined in [14], yet, *there is no cryptographic primitive for $\mathcal{A}$ to break at all*! Without a cryptographic primitive (which is computationally hard) embedded, the proof is destined to fail.

Let us investigate into their reasoning of the proof and see where the flaw lies. Li *et al.* defined Exp_3 as "if $\mathcal{A}$ can correctly guess SID, V_1, H_{MAC}, S_2 and V_2, then the connection is aborted", and stated that $\Delta_2 = |\Pr[Exp_3] - \Pr[Exp_2]| \leq \frac{q_{send}}{|\mathcal{D}_{pw}|}$, where q_{send} stands for the number of on-line attacks launched by $\mathcal{A}$. However, the user anonymity violation attack described in Sec. III implies that $\Delta_2 = |\Pr[Exp_3] - \Pr[Exp_2]| = 1$ which is non-negligible, because once $\mathcal{A}$ has obtained MU's identity ID_{MU}, she can compute $sk_{MU} = h(h(ID_{MU}\|ID_{HA})\oplus h(x)\oplus x_0 \oplus y_0))$, where ID_{HA}, x_0, y_0 are intercepted from the open channel and $h(x)$ is extracted from any legitimate user's (or $\mathcal{A}$'s own) smart card. Therefore, the proof is invalid. Similarly, Li *et al.*'s proof for user anonymity fails, too.

Remark 3. Once again, our attacks demonstrate that having a formal security model and designing a "provably secure" protocol in that model is no panacea for assuring security. In the case of Li *et al.*'s work, the failure of their proof is mainly due to the lack of a rigorous security reduction. Actually, either a non-tight security reduction, though which are reasoned in a well defined model, or an insufficient security model, which do not capture all the realistic capabilities of the adversary, can render the seemingly sound proof entirely meaningless in practice. Many schemes are armed with a formal model, but they fail to achieve the claimed security goals and need to be fixed [25]–[27]. This highlights the important role that old-fashioned cryptanalysis continues to play in establishing confidence in the security of a cryptographic protocol, and suggests that special attention shall be given to the tightness of reduction argument.

B. An improved scheme

The inherent flaw revealed in Li *et al.*'s security proof supports "the public-key principle" [21] that two-factor protocols, which do not involve public-key operations but assume the non-tamper resistance of smart cards, are intrinsically unable to resist offline password guessing attack. Consequently, the right way to tackle the identified defects in [14] is to resort to public-key techniques. This will unavoidablely lose some efficiency.

Since the home/foreign agents (e.g., access points) are comparatively more powerful than the smart cards used by the mobile users, the mobile networks are typically imbalanced networks. Inspired by the work of Zhu *et al.* [28], we proposed an enhancement that exploits the computational imbalance of RSA encryption technique, in which the *encryption* operation on the user side usually incurs much less computation cost than that of the *decryption* operation on the home/foreign agents, to enhance protocol security while maintaining acceptable efficiency for the mobile devices. The notations used are listed in Table I.

1) Registration phase: The home agent HA generates two large primes p and q and computes $n = pq$, then selects a prime number e and an integer d, such that $ed = 1 \bmod (p-1)(q-1)$. Finally, HA publishes n and e, while p, q and d are kept secret. Generally, e is recommended to be a small value, e.g. $e = 3$ or 65537 [29] to increase efficiency. Whenever a mobile user MU enrolls in HA, the following steps are involved:

(1) MU chooses her identity ID_{MU}, password PW_{MU}.
(2) $MU \Rightarrow HA : \{ID_{MU}, PW_{MU}\}$.
(3) HA selects a random number $X_{MU} \in_R \mathbb{Z}_p^*$, sets T_{reg} to be the current timestamp and computes $K_{MU} = h(ID_{MU} \| ID_{HA} \| X_{MU} \| h(d) \| T_{reg})$, $u = PW_{MU} \oplus K_{MU}$. Then HA stores $(ID_{MU}, X_{MU}, T_{reg})$. Note that, T_{reg} is used for revoking lost smart cards and HA chooses the value of X_{MU} corresponding to each user to make sure K_{MU} is unique for each user.
(4) $HA \Rightarrow MU$: A smart card with data $\{ID_{HA}, u, n, e\}$.

2) Login phase: When MU roams into a foreign network managed by a foreign agent FA, she authenticates herself to FA to show that she is a subscriber of her home agent HA.

(1) MU inserts her smart card into a card reader and inputs her identity ID_{MU}, password PW_{MU}.
(2) The smart card generates a random number $r \in_R \mathbb{Z}_p^*$, reads current timestamp t_{MU}, and computes $x_0 = r^e \bmod n$, $K_{MU} = u \oplus PW_{MU}$, $SID = ID_{MU} \oplus h(r \| t_{MU})$ and $V_1 = h(SID \parallel K_{MU} \parallel r \parallel t_{MU})$.
(3) $MU \rightarrow FA : m_1 = \{SID, V_1, ID_{HA}, x_0, t_{MU}\}$.

3) Mutual authentication phase:

(1) On receiving m_1, FA checks the freshness of t_{MU} by checking $t_{FA} - t_{MU} \leq \triangle T_1$, where t_{FA} is the current time of FA and ∇T_1 is the permissible time interval for the transmission delay between FA and its users. If t_{MU} is invalid, FA rejects.
(2) FA generates a random number y_0, computes $H_{mac} = h(y_0 \parallel ID_{FA} \parallel K_{FH} \parallel V_1 \| t_{FA} \| t_{MU})$, where K_{FH} is a pre-shared symmetric key between FA and HA.
(3) $FA \rightarrow HA : m_2 = \{m_1, y_0, H_{mac}, ID_{FA}, t_{FA}\}$.
(4) Upon receiving m_2, HA performs the following steps:
   a) HA checks the freshness of t_{FA} by checking $t_{HA} - t_{FA} \leq \triangle T_2$, where t_{HA} is the current time of HA and ∇T_2 is the permissible time interval for a transmission delay between HA and the foreign agents. If t_{FA} is invalid, HA rejects.
   b) HA retrieves K_{FH} according to ID_{FA} and checks $H_{mac} \stackrel{?}{=} h(y_0 \parallel ID_{FA} \parallel K_{FH} \parallel V_1 \| t_{FA} \| t_{MU})$. If they are not equal, HA terminates.
   c) HA decrypts x_0 to obtain r, computes $ID_{MU} = SID \oplus h(r \| t_{MU})$, and retrieves X_{MU}, T_{reg} according to ID_{MU} from its backend database. HA computes $K_{MU} = h(ID_{MU} \| ID_{HA} \| X_{MU} \| h(d) \| T_{reg})$ and checks whether $V_1 \stackrel{?}{=} h(SID \parallel K_{MU} \parallel r \parallel t_{MU})$. If they are not equal, HA rejects.
   d) HA computes $V_2 = h(ID_{MU} \parallel ID_{HA} \parallel ID_{FA} \| r \| t_{HA})$, $E_1 = E_{K_{FH}}(h(r \| y_0))$, and $S_2 = h(ID_{FA} \| V_2 \| E_1 \| t_{HA})$.
(5) $HA \rightarrow FA : m_3 = \{S_2, V_2, E_1, t_{HA}\}$.
(6) After FA receives message m_3 , it checks the freshness of t_{HA} by checking $t'_{FA} - t_{HA} \leq \triangle T_2$, where t'_{FA} is current time of FA, and verifies $S_2 \stackrel{?}{=} h(ID_{FA} \| V_2 \| E_1 \| t_{HA})$. If either check fails, FA terminates. Otherwise, FA computes the session key $sk_{FA} = h(h(r \| y_0) \| t_{MU} \| t'_{FA})$ and generates a temporary certificate $TCertu$ for MU.
(7) $FA \rightarrow MU : m_4 = \{y_0, E_{sk_{FA}}(TCertu \parallel V_2 \parallel ID_{FA}), t'_{FA}\}$.
(8) On receiving message m_4, MU checks the freshness of t'_{FA} by checking $t'_{MU} - t'_{FA} \leq \triangle T_1$, where t'_{MU} is current time of MU. MU computes the session key $sk_{MU} = h(h(r \| y_0) \| t_{MU} \| t'_{FA})$, decrypts $E_{sk_{FA}}(TCertu \parallel V_2 \parallel ID_{FA})$ with sk_{MU} and checks whether V_2 is equal to $h(ID_{MU} \parallel ID_{HA} \parallel ID_{FA} \parallel r \| t_{HA})$. The equality indicates the successful mutual authentication between MU, FA and HA.

4) Session key update phase and password change phase: The session key update phase is the same with that of Li *et al.*'s scheme. As for the password change phase, we emphasize that MU has to interact with HA to update her password, for a recent study [21] points out that schemes that enable securely local password update are definitely vulnerable to offline password guessing attack. Hence, in the improved scheme, MU needs first to be authenticated by HA by performing the login phase and authentication phase, then the password can be changed by updating u with $u \oplus PW_{MU}^{old} \oplus PW_{MU}^{new}$ in the card memory.

V. Security analysis and efficiency evaluation

A. Security analysis

As the improved scheme is based on Li *et al.*'s scheme, we only demonstrate that it can meet the missing security goals.

Proposition 1: Our scheme can achieve user anonymity.

Proof: Firstly, let us show that user identity-protection can be preserved. There are only two protocol transcripts, i.e. SID and V_2, that contain the information about ID_{MU}. In the login request m_1, ID_{MU} is concealed in $SID = ID_{MU} \oplus h(r \| t_{MU})$, where $r = x_0^d \bmod n$. As only HA knows the private key d, there is no way for $\mathcal{A}$ to recover ID_{MU} from SID. Besides, deriving ID_{MU} from $V_2 = h(ID_{MU} \parallel ID_{HA} \parallel ID_{FA} \| r \| t_{HA})$ is also intractable since $h(\cdot)$ is a secure hash function (e.g. SHA-1 [30]).

Secondly, let us show that user un-traceability can be preserved. Suppose that $\mathcal{A}$ has intercepted all the protocol transcripts $\{m_1, m_2, m_3, m_4\}$ and extracted the parameters $\{ID_{HA}, u, n, e\}$ in her own smart card. Then, $\mathcal{A}$ may try to retrieve any user-specific static element from these transcripts to identify MU, but m_1, m_2, m_3, and m_4 are all session-variant and indeed random strings due to the randomness of r. Accordingly, without knowledge of the random number r, the adversary will have to solve the large integer factorization problem to retrieve the correct value of ID_{MU} from SID or K_{MU} from V_1, while ID_{MU} and K_{MU} are the only user-specific static elements in the protocol transcripts. ■

Proposition 2: Our scheme can withstand offline password guessing attack even if the victim's smart card is lost.

Proof: Suppose that $\mathcal{A}$ has intercepted MU's protocol transcripts $\{m_1, m_2, m_3, m_4\}$ and extracted the data $\{ID_{HA}, u, n, e\}$ in MU's lost smart card. Even after gathering such information, without the knowledge of X_{MU} or d, $\mathcal{A}$ has to either break the hash function $h(\cdot)$ or solve the large integer factorization problem to guess PW_{MU}. In this way, our scheme resists against offline password guessing attack. ■

TABLE III
PERFORMANCE COMPARISON AMONG RELATED SCHEMES

Related schemes	Computation cost on user side	Resistance to known attacks	User anonymity
Li *et al.*(2013) [14]	$5T_H + 1T_S$	×	×
Isawa-Morii(2012) [22]	$7T_H + 3T_S$	× [21]	×
He *et al.*(2011) [4]	$10T_H + 2T_S$	× [31]	×
Zhou-Xu(2011) [12]	$2T_E + 5T_H + 2T_S$	× [14]	✓
Xu *et al.*(2011) [13]	$2T_E + 3T_H + 1T_S$	× [31]	✓
Our improved scheme	$1T_{SE} + 4T_H + 1T_S$	✓	✓

Note: $\times[x]$ means the corresponding scheme fails to meet some security goal(s), and the defect(s) is uncovered by the reference [x]. The schemes in [4], [22] are prone to a similar user anonymity violation attack as described in Sec. III.

B. Performance evaluation

We mainly evaluate the efficiency of the login and authentication phases, since these two phases are the main part of an authentication scheme and are executed much more frequently than the other phases. The results about the cryptographic primitives that are involved in these two phases of our scheme and other five latest schemes [4], [4], [13], [14], [22] are summarized in Table III, where T_E, T_{SE}, T_H, T_S denote the time complexity for exponentiation, small-exponent exponentiation, hash function, and symmetric encryption/decryption, respectively. Other lightweight operations (e.g., XOR) are not taken into account. Particularly, we focus on the numbers of cryptographic operations that MU needs to perform, because mobile devices usually are low-powered devices and thus computation cost at the user end is always regarded as a key criteria.

It is worth noting that the small-exponent exponentiation operation is much less costly as compared to the common exponentiation operation, i.e. $T_{SE} << T_E$. For example, Scott *et al.* [32] reported that, when the small-exponent e is set to $2^{16}+1$ (and $|n| = 1024$ bit), one such small-exponent exponentiation only takes 5.384 milliseconds on a 32-bit 36MHz RISC MIPS-based smart card, while one common exponentiation costs 0.14 seconds. In a nutshell, our scheme maintains reasonable efficiency and is well suited to mobile environments.

VI. CONCLUSION

In this paper, we have revisited Li *et al.*'s scheme and shown that, despite being equipped with a claimed proof of provable security, this scheme fails to preserve user anonymity and suffers from offline password guessing attack. To explicate this seemingly paradoxical situation, we have investigated into Li *et al.*'s security proof and uncovered the flaw in the reasoning of the proof. We further presented an improved scheme to cope with the identified security defects without sacrificing any desirable feature of the original scheme. We conjecture that the strategy of only using symmetric-key techniques to achieve user anonymity is intrinsically infeasible, and pose its rigorous justification as one interesting (and fundamental) open problem.

REFERENCES

[1] H. Shirazi, J. Cosmas, and D. Cutts, "A cooperative cellular and broadcast conditional access system for pay-tv systems," *IEEE Trans. Multimedia*, vol. 56, no. 1, pp. 44–57, 2010.

[2] A. G. Vicente, I. B. Munoz, J. L. L. Galilea, and P. A. R. del Toro, "Remote automation laboratory using a cluster of virtual machines," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3276–3283, 2010.

[3] X. Li, W. Qiu, K. Chen, and J. Li, "Anonymity enhancement on robust and efficient password-authenticated key agreement using smart cards," *IEEE Trans. Ind. Electron.*, vol. 57, no. 2, pp. 793–800, 2010.

[4] D. He, M. Ma, Y. Zhang, C. Chen, and J. Bu, "A strong user authentication scheme with smart cards for wireless communications," *Comput. Commun.*, vol. 34, no. 3, pp. 367–374, 2011.

[5] G. M. Yang, D. S. Wong, H. X. Wang, and X. T. Deng, "Two-factor mutual authentication based on smart cards and passwords," *J. Comput. Syst. Sci.*, vol. 74, no. 7, pp. 1160–1172, 2008.

[6] J. Zhu and J. Ma, "A new authentication scheme with anonymity for wireless environments," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 231–235, 2004.

[7] D. Wang and C.-G. Ma, "Cryptanalysis of a remote user authentication scheme with provable secuirty for mobile client-server environment based on ECC," *Inform. Fusion*, vol. 14, no. 4, pp. 498–503, 2013.

[8] C.-C. Lee, M.-S. Hwang, and I.-E. Liao, "Security enhancement on a new authentication scheme with anonymity for wireless environments," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1683–1687, 2006.

[9] C.-C. Wu, W.-B. Lee, and W.-J. Tsaur, "A secure authentication scheme with anonymity for wireless communications," *IEEE Commun. Lett.*, vol. 12, no. 10, pp. 722–723, 2008.

[10] P. Zeng, Z. Cao, K.-k. Choo, and S. Wang, "On the anonymity of some authentication schemes for wireless communications," *IEEE Commun. Lett.*, vol. 13, no. 3, pp. 170–171, 2009.

[11] C.-C. Chang, C.-Y. Lee, and Y.-C. Chiu, "Enhanced authentication scheme with anonymity for roaming service in global mobility networks," *Comput. Commun.*, vol. 32, no. 4, pp. 611–618, 2009.

[12] T. Zhou and J. Xu, "Provable secure authentication protocol with anonymity for roaming service in global mobility networks," *Comput. Netw.*, vol. 55, no. 1, pp. 205–213, 2011.

[13] J. Xu, W.-T. Zhu, and D.-G. Feng, "An efficient mutual authentication and key agreement protocol preserving user anonymity in mobile networks," *Comput. Commun.*, vol. 34, no. 3, pp. 319–325, 2011.

[14] H. Li, Y. Yang, and L. Pang, "An efficient authentication protocol with user anonymity for mobile networks," in *Proc. WCNC 2013*. IEEE, 2013, pp. 1842–1847.

[15] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, 2002.

[16] T. H. Kim, C. Kim, and I. Park, "Side channel analysis attacks using am demodulation on commercial smart cards with seed," *J. Syst. Soft.*, vol. 85, no. 12, pp. 2899 – 2908, 2012.

[17] J. Bonneau, M. Just, and G. Matthews, "Whats in a name?" in *Proc. FC 2010*, ser. LNCS, R. Sion, Ed. Springer, 2010, vol. 6052, pp. 98–113.

[18] M. Dell'Amico, P. Michiardi, and Y. Roudier, "Password strength: An empirical analysis," in *Proc. INFOCOM 2010*, march 2010, pp. 1–9.

[19] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Proc. EUROCRYPT 2000*, ser. LNCS, B. Preneel, Ed. Springer-Verlag, 2000, vol. 1807, pp. 139–155.

[20] "Miracl library," Shamus Software Ltd., June 2013, http://www.shamus.ie/index.php?page=home.

[21] C.-G. Ma, D. Wang, and S. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *Int. J. Commun. Syst.*, 2012, doi: http://dx.doi.org/10.1002/dac.2468.

[22] R. Isawa and M. Morii, "Anonymous authentication scheme without verification table for wireless environments," *IEICE Trans. Fund. Electron. Commun. Comput. Sci.*, vol. 95, no. 12, pp. 2488–2492, 2012.

[23] M.-C. Chuang, J.-F. Lee, and M.-C. Chen, "Spam: A secure password authentication mechanism for seamless handover in proxy mobile ipv6 networks," *IEEE Syst. J.*, vol. 7, no. 1, pp. 102–113, 2013.

[24] E. Bresson, O. Chevassut, and D. Pointcheval, "Security proofs for an efficient password-based key exchange," in *Proc. ACM CCS 2003*. New York, NY, USA: ACM, 2003, pp. 241–250.

[25] G. Yang, "Comments on an anonymous and self-verified authentication with authenticated key agreement for large-scale wireless networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 6, pp. 2015–2016, 2011.

[26] D. He, C. Chen, S. Chan, and J. Bu, "Analysis and improvement of a secure and efficient handover authentication for wireless networks," *IEEE Communi. Lett.*, vol. 16, no. 8, pp. 1270–1273, 2012.

[27] G. Wang, J. Yu, and Q. Xie, "Security analysis of a single sign-on mechanism for distributed computer networks," *IEEE Trans. Ind. Inform.*, vol. 9, no. 1, pp. 294–302, 2013.

[28] F. Zhu, D. Wong, A. Chan, and R. Ye, "Password authenticated key exchange based on RSA for imbalanced wireless networks," in *Proc. ISC 2002*, ser. LNCS. Springer-Verlag, 2002, vol. 2433, pp. 150–161.

[29] "Public-key cryptography standards, PKCS#11 mechanisms v2.30," RSA Security Inc., http://www.rsasecurity.com/rsalabs/pkcs/.

[30] "Fips pub 180-4: Secure hash standard," NIST, Mar. 2012, http://www.nist.gov/customcf/get_pdf.cfm?pub_id=910977.

[31] S. Wu, Y. Zhu, and Q. Pu, "A novel lightweight authentication scheme with anonymity for roaming service in global mobility networks," *Int. J. Netw. Manag.*, vol. 21, no. 5, pp. 384–401, 2011.

[32] M. Scott, N. Costigan, and W. Abdulwahab, "Implementing cryptographic pairings on smartcards," in *Proc. CHES 2006*, ser. LNCS, L. Goubin and M. Matsui, Eds. Springer-Verlag, 2006, vol. 4249, pp. 134–147.