On the Challenges in Designing Identity-based Privacy-Preserving Authentication Schemes for Mobile Devices

Ding Wang *Student Member, IEEE*, Haibo Cheng, Debiao He and Ping Wang *Senior Member, IEEE*

***Abstract*—Providing secure, efficient and privacy-preserving user authentication in mobile networks is a challenging problem due to the inherent mobility of users, variety of attack vectors and resource-constrained nature of user devices. Recent studies show that identity-based cryptosystems can eliminate the certificate overhead and thus address the issues associated with public key infrastructure (PKI) technology—which is a rare bit of good news in today's computer security world. In this work, we employ three representative identity-based remote user authentication schemes (i.e., Truong et al.'s scheme, Li et al.'s scheme and Zhang et al.'s scheme) as case studies to *reveal the challenges and subtleties* in designing a practical authentication scheme for mobile devices.**

First, we demonstrate that Truong et al.'s scheme, which was presented at AINA'12, cannot achieve the *claimed* security goals, and we report its following flaws: (1) it still fails to resist against known session-specific temporary information attack; (2) it cannot withstand key compromise impersonation attack; (3) it is of poor usability. Second, we show that Li et al.'s privacy-preserving scheme, which was proposed at GLOBECOM'12, is subject to some subtle (yet severe) efficiency problems that make it virtually impossible for any practical use. Third, we scrutinize a "provably secure" scheme for roaming services in mobile networks designed by Zhang et al. at SCN'15, and find it prone to collusion attack and replay attack. Further, we *investigate into the underlying causes* for these identified failures and figure out an improvement over Truong et al.'s scheme to overcome the revealed challenges while maintaining reasonable efficiency.

keywords–Mobile authentication, Privacy-preserving, ID-based cryptography, Cryptanalysis, User anonymity.

I. Introduction

With the rapid development of wireless technologies (e.g., GSM, GPRS, WiMAX, LTE, Zigbee, VLC) and the proliferation of various mobile devices (e.g., personal digital assistants, notebook PCs, sensors, smart phones and wearable devices), pervasive Internet access is becoming a reality, enabling mobile subscribers to enjoy comprehensive services offered by various applications [1], [2] at anytime and anywhere. However, this emerging paradigm of networking raises prominent new challenges [3], [4] in the security of systems and the privacy of mobile users.

Many applications handle user personal sensitive information, such as their locations and movements, or their health status and purchasing preferences, and thus it is of great concern to protect the systems and the users' privacy and security from malicious adversaries. Whenever a mobile user wants to login a remote service provider (which may be a powerful cloud server [5] or a lightweight sensor node [6]) and access the desired data/services, such as e-health, home automation, Internet banking and pay-TV, both the user and the service provider (which we hereafter call "server" for short) must validate the authenticity of the corresponding party by acquirement of corroborative evidence.

D. Wang, H. Cheng and P. Wang are with the School of EECS, Peking University, Beijing, China. Email: {wangdingg, hbcheng, pwang}@pku.edu.cn
D. He is with the School of Computer Science, Wuhan University, Wuhan 430072, P.R. China. Email: hedebiao@163.com

To provide mutual authentication between the user and the server, a great number of user authentication schemes have been proposed, including the famous "Kerberos" [7], "HMQV" [8] and "NAXOX" [9]. However, these traditional remote user authentication schemes rely on the intractability of the large integer factoring problem, computational Diffie-Hellman problem and their variants. In other words, they are based on public-key cryptosystems (PKC), such as RSA and ElGamal. However, PKC needs to compute the time-consuming modular exponential operations. In addition, the PKC-based schemes need an extra key management system for certificate control [10], [11]. Since the computational ability, memory and battery capacity of mobile devices are often very limited, the traditional PKC-based authentication schemes are unsuitable for applications where mobile devices are used.

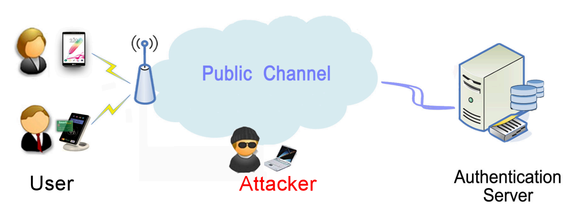

Fig. 1. User authentication for mobile devices

Compared with traditional PKC, ID-based cryptography (IBC) exploits an entity's ID or email address as her public key and thus completely eliminates the expensive management cost of public key certificates, which is particularly desirable in mobile environments [12]. In addition, IBC is often implemented by an elliptic curve to offer better performance, because computation in an elliptic curve can achieve the same security strength by using a much smaller key size as compared to that of the finite field. For example, 163-bit ECC and 1024-bit RSA have the same security level in practice [13]. Thus, ID-based authentication schemes show great advantages for mobile application scenarios where low-weight devices with restricted resources are involved.

In 2009, Yang and Chang [14] proposed an ID-based scheme for mobile user authentication based on ECC. Although Yang-Chang's scheme keeps merits of both the elliptic curve and ID-based cryptosystems and is more superior in terms of efficiency than most of the previous ones, in 2011, Islam and Biswas [15] showed that the Yang-Chang's scheme [14] suffers from a number of issues such as known session-specific temporary information attack, and is short of user anonymity and forward secrecy. To remedy these security flaws, they proceeded to propose a new ID-based scheme.

At AINA'12, Truong et al. [16] found that Islam-Biswas's scheme [15] is vulnerable to denial of service attack and known session-specific temporary information attack, and presented an improvement over Islam-Biswas's scheme. Truong et al. claimed that their improved scheme provides mutual authentication and is free from all known cryptographic attacks such as replay attack, impersonation attack and so on. Their scheme is superior in efficiency to the previous solutions for implementation on mobile devices, yet as we will show in this work, it cannot achieve the claimed security goals.

In 2012, He et al. [17] pointed out that it is still an open issue to ensure security and efficiency in the process of seamless handover over multiple access points for mobile nodes, and proposed a novel ID-based handover authentication scheme. However, at GLOBECOM'12, Li et al. [18] revealed that He et al.'s scheme is short of forward secrecy and key privacy and moreover, it involves the computation of a number of bilinear pairings and thus the computation cost may not be satisfactory. Accordingly, Li et al. [18] suggested a new ID-based scheme without using any bilinear pairing operations while preserving user privacy and withstanding various known attacks. Thus, Li et al.'s scheme [18] shows great advantages over existing related schemes. Nevertheless, in this work we will reveal that, user anonymity of this scheme is achieved at great cost of management and communication overhead, which makes the scheme hardly suitable for practical use.

In 2015, Zhang et al. [19] investigated into the issues in designing a secure and efficient authentication scheme for roaming services in mobile networks. In such network environments, there are three entities involved, i.e., a mobile user (MU), a home server (HS) and a foreign server (FS). To gain ubiquitous network access regardless of her location, a MU need to authenticate to a FS with the help of her HS; To prevent users' location and activities from being tracked by attackers or curious FSs, it is essential to preserve user anonymity. Accordingly, Zhang et al. [19] advanced a new scheme to eliminate the defects in existing schemes in [20], [21] by using ID-based cryptography. However, we find that though this scheme is equipped with a formal proof in the random oracle model, it is prone to a new kind of attack —collusion attack, in which an attacker colludes with a *legitimate yet curious* foreign server can offline guessing MU's password. Besides, we point out that a number of recent schemes (e.g., [22], [23]) cannot withstand this attack. While all the schemes in [19], [22], [23] have been "proved secure" in some formal model, our results highlight that formal method is no panacea for assuring security and it is critical to be aware of potential threats when designing a cryptographic protocol. This suggests the necessity of this work.

In a nutshell, this paper makes four main contributions:

(1) First, we demonstrate that Truong et al.'s scheme actually cannot achieve the claimed security goals and is vulnerable to known session-specific temporary information attack and key compromise impersonation attack. In addition, the users in their scheme have to remember a random authentication key, which renders the scheme user-*un*friendly.

(2) Second, we reveal an inherent design weakness in Li et al.'s scheme. For n users in the system, every home/foreign authentication server shall maintain a list of all pseudo-IDs that have been used by these n users. This implies that, once a pseudo-ID has been utilized by a user, this information shall be signaled to all the other servers in the system, otherwise this pseudo-ID along with the corresponding credentials can be replayed. This means a broadcast flooding occurs, rendering the scheme hardly usable.

(3) Third, we, for the first time, identify a new effective attack, in which an external attacker colludes with a curious foreign server to guess the user's password, on roaming authentication schemes. We use one of the foremost scheme (i.e., Zhang et al.'s scheme) as a case study to show its damaging threat. Particularly, our cryptanalysis results on this "provably secure" scheme once again underline the crucial role of old-fashioned cryptanalysis and the importance of being aware of potential threats when designing a protocol.

(4) Last but not the least, we figure out the roots of the identified failures in these three schemes and put forward effective countermeasures to fix the security and usability problems in Truong et al.'s scheme without losing any features, while we find the other two schemes cannot be amended with moderate revisions.

The remainder of this paper is organized as follows: in Sec. II, we review Truong et al.'s scheme. Sec. III describes the weaknesses of Truong et al.'s scheme. Sec. IV reveals the subtle efficiency problem in Li et al.'s scheme. Sec. V highlights the feasibility of collusion attack Zhang et al.'s scheme. The corresponding remedies for Truong et al.'s scheme are given is Sec. VI. Sec. VII concludes the paper.

II. Review of Truong et al.'s scheme

In this section, we review the ID-based remote user authentication scheme for mobile users based on ECC proposed by Truong et al. [16]. We are only interested in the first three phases of this scheme: system initialization, registration, authentication and session key agreement. For ease of presentation, we list some intuitive notations in Table I.

A. The system initialization phase

Before the system begins, server S performs as follows:

Step S1. S selects a k-bit prime number p and a base point $\mathcal{P}$ with order n from the elliptic curve group G_p.

TABLE I
NOTATIONS AND ABBREVIATIONS

Symbol	Description
U_i	i^{th} user
S	remote server
$\mathcal{A}$	the adversary
ID_i	identity of user U_i
$\oplus$	the bitwise XOR operation
$\parallel$	the string concatenation operation
$H(\cdot)$	collision free one-way hash function
q_s	secret key of remote server S
$\mathcal{O}$	the point at infinity
$\mathcal{P}$	base point of the elliptic curve group of order n such that $n \cdot \mathcal{P} = \mathcal{O}$
$\mathcal{Q}_s$	public key of server S, where $\mathcal{Q}_s = q_s \cdot \mathcal{P}$
$\rightarrow$	an open communication channel
$\Rightarrow$	a secure communication channel

Step S2. S selects a random number q_s from $[1, n-1]$ as its secret key, and chooses three one-way secure hash functions: $H_1 : \{0,1\}^* \rightarrow G_p$, $H_2 : G_p \times G_p \rightarrow \{0,1\}^k$ and $H_3 : G_p \rightarrow \{0,1\}^k$.

Step S3. S publishes $\{E_p(a,b), \mathcal{P}, H_1(\cdot), H_2(\cdot), H_3(\cdot)\}$.

B. The registration phase

The registration phase involves the following operations:

Step R1. U_i chooses her identity $ID_i \in_R \{0,1\}^k$.

Step R2. $U_i \Rightarrow S : \{ID_i\}$.

Step R3. On receiving the registration message from U_i, the server S checks U_i's identity. If ID_i already exists in the server's database, S asks U_i for a different identity. Then, S computes the authentication key $AID_i = q_s \cdot H_1(ID_i \parallel X_i)$, where X_i is a random number chosen for U_i from $[1, p-1]$. S creates an entry (ID_i, `status-bit`) in its database, where the `status-bit` indicates the status of the client, i.e., when the client is logged-in to the server the `status-bit` is set to one, otherwise it is set to zero.

Step R4. $S \Rightarrow U_i$: security parameters $\{AID_i, X_i\}$.

C. Login and authentication phase

When U_i wants to login to S, U_i performs:

Step L1. U_i keys her identity ID_i and her long-term key AID_i into the mobile device, the mobile device chooses a number $r_u \in_R [1, n-1]$, and computes $R_i = r_u \cdot H_1(ID_i || X_i)$, $R' = r_u \cdot AID_i$, $M_i = H_2(R' \parallel AID_i)$ and $CID_i = ID_i \oplus H_3(R')$.

Step L2. $U_i \rightarrow S : \{X_i, CID_i, M_i, R_i\}$

Step L3. Upon receiving the login request from U_i, S computes $R'^* = q_s \cdot R_i$, $ID_i^* = CID_i \oplus H_3(R'^*)$, and then checks the validity of the identity ID_i^*. If ID_i^* is valid, S go to the next step, otherwise rejects the login request.

Step L4. S computes the authentication key $AID_i^* = q_s \cdot H_1(ID_i^* \parallel X_i)$ and checks whether $H_2(R'^* \parallel AID_i^*)$ equals the received M_i. If it doesn't hold, S rejects U_i's login request, otherwise chooses a random number r_s from $[1, n-1]$. Then, S computes $R_s = r_s \cdot AID_i^*$, $T_s = R'^* + R_s$ and $H_s = H_2(R_s \parallel AID_i^*)$.

Step L5. $S \rightarrow U_i : \{T_s, H_s\}$.

Step L6. U_i computes $R_s^* = T_s - R'$ and $H_s^* = H_2(R_s^* \parallel AID_i)$, and rejects if H_s^* is unequal to the received H_s. Then U_i computes $H_{RS} = H(R' \parallel R_s^*)$ and the session key $SK = H_3(r_i \cdot R_s^*)$.

Step L7. $U_i \rightarrow S : \{H_{RS}\}$.

Step L8. The server S computes $H_{RS}^* = H_2(R'^* \parallel R_s)$ and compares H_{RS}^* with the received H_{RS}. If the equality holds, S grants the client's login request and computes the session key $SK = H_3(r_s \cdot R_i^*)$, otherwise rejects.

III. Cryptanalysis of Truong et al.'s scheme

With superior performance over other related schemes and a long list of arguments of admired features (such as user anonymity and device revocation) that their scheme possesses presented, Truong et al.'s scheme [16] seems quite promising from the prospective of desirable features. However, without investigation into the underlying (fundamental) causes of previous security failures, its security analysis is highly to be problematic and as we will show, this scheme still fails to serve its purposes by demonstrating its vulnerabilities to concrete yet realistic attacks.

A. Known session-specific temporary information attack

As noted by Canetti and Krawczyk [24], the known session-specific temporary information attack concerns with the damage of leakage of ephemeral secrets in a specific protocol run. A protocol is said to resist against this kind of attack if it can ensure that, the disclosure of information specific to one session (such as the leakage of the session key or ephemeral state information during the protocol run) has no effects on the security of other sessions.

Truong et al. [16] pointed out that Islam-Biswas's scheme [15] cannot provide resistance against known session-specific temporary information attack and made an effort to overcome this vulnerability. Although Truong et al. claimed that their scheme has thwarted this threat, the following attacking procedure will be given here as a counterexample.

Let us consider the following scenarios. In case the ephemeral exponent r_u accidentally is somehow obtained (e.g., accidental leakage or intentionally stolen) by an adversary $\mathcal{A}$. Once the login request message $\{X_i, CID_i, M_i, R_i\}$ during any authentication process is intercepted by $\mathcal{A}$, $\mathcal{A}$ can obtain U_i's identity as follows:

Step 1. Guesses the value of ID_i to be ID_i^* from a uniformly distributed identity dictionary $\mathcal{D}_{id}$.

Step 2. Computes $R_i^* = r_u \cdot H_1(ID_i^* \parallel X_i)$, where X_i is intercepted from the open channel.

Step 3. Verifies the correctness of ID_i^* by checking if the computed R_i^* equals the intercepted R_i.

Step 4. Repeats Steps 1, 2 and 3 of this procedure until the correct value of ID_i is found.

Note that, the above attack is very effective, because it only requires the capabilities of an eavesdropping, passive guessing attacker, and involves no special cryptographic operations. Let $|\mathcal{D}_{id}|$ denote the size of the identity dictionary $\mathcal{D}_{id}$. The time complexity of the above attack procedure is $\mathcal{O}(|\mathcal{D}_{id}| * (T_P + T_H))$, where T_P is the running time for point multiplication and T_H is the running time for Hash function. That is, the time for $\mathcal{A}$ to recover U_i's identity is a linear function of the identity dictionary size $|\mathcal{D}_{id}|$.

In Truong et al.'s scheme, the user is allowed to select her identity ID_i at will during the registration phase. As is well known, users usually tend to select an identity that is easy to remember for their convenience, or some phrases that are meaningful (e.g., related to themselves, family members, relatives or favorite band names) [25]. Hence, the space of $\mathcal{D}_{id}$ shall be very limited in practice, e.g., $|\mathcal{D}_{id}| \leq |\mathcal{D}_{pw}| \leq 10^6$ [26], [27]. To further show the effectiveness of our attack, we make use of the publicly-available, rational arithmetic C/C++, cryptographic library MIRACL [28], and implement the ECC point multiplication operation and Hash operation on common PCs and attain the corresponding operation timings (see Table II). For example, one T_P operation and one T_H only take about 1.186 ms and 0.631 μs on a common Intel i5-2450M 2.50 GHz processor, respectively. In all, the above attack procedure can be completed in about 20 minutes on a common PC.

TABLE II
COMPUTATION EVALUATION OF RELATED OPERATIONS ON LAPTOP PCS

Experimental Platform (common PCs)	ECC Point Multiplication T_P(sect163r1 [29])	Symmetric enc. T_S (AES-128)	Hash operation T_H (SHA-1)
Intel E5500 2.80 GHz	0.615 ms	0.530 µs	0.739 µs
Intel i5-3210 2.50 GHz	0.942 ms	0.457 µs	1.106 µs
Intel i5-2450M 2.50 GHz	1.186 ms	0.152 µs	0.631 µs

Once the correct value of ID_i is obtained by $\mathcal{A}$, user anonymity (privacy) will be violated, while user anonymity is an important feature that a practical mobile authentication scheme should achieve [30], [31] and is also a primary goal of Truong et al.'s scheme. Hence, this scheme is vulnerable to known session-specific temporary information attack.

B. Key compromise impersonation attack

In the case of key compromise impersonation (KCI) [8], an adversary $\mathcal{A}$ is assumed to be with the knowledge of a communicating party i's private key, and this enables $\mathcal{A}$ not only to masquerade party i to others but also to masquerade other *un*compromised parties $j(j \neq i)$ to i. Schemes free from such reverse impersonation are considered to be able to meet the security goal of resistance against KCI attack.

We illustrate that resistance to KCI attack for mobile authentication protocols is as crucial as the other security goals, when considering the recent endless catastrophic leakages of user-sensitive services (e.g., Xiaomi cloud and Evernote cloud [32]) and the prevalence of zero-day attacks like "Heartbleed" [33]. To this end, we show the following typical scenario where a KCI attack is really damaging. In a cloud-based file sharing system, each user U's mobile phone can assess her private data (e.g., photos and videos) that is stored on the remote cloud server S. The access of U's private data stored in S is *only* allowed to the single entity U (while the data even cannot be accessed by the cloud server for privacy reasons, which is quite realistic in practice.). This can be fulfilled by executing a authentication protocol between U and S and after successful authentication, S sends the data encrypted by using the agreed session key. The goal of an adversary $\mathcal{A}$ is to access the data stored at S (note that, this data can be shared only with user U who has read access). Though the compromise of S's long-term key helps $\mathcal{A}$ to impersonate S, $\mathcal{A}$ may be unable to gain the data locally stored at S because of access control privileges. However, if the underlying protocol used is prone to KCI attacks, $\mathcal{A}$ can impersonate U who has read access and decrypt the data using the session key.

Assume the long-term secret key q_s of the server S has somehow been learned (e.g., through Zero-day attacks) by the adversary $\mathcal{A}$. Without loss of generality, we suppose one of U_i's previous login requests, say $\{X_i, CID_i, M_i, R_i\}$, is intercepted by $\mathcal{A}$. Once the value of q_s is obtained, with the previously intercepted protocol transcripts $\{X_i, CID_i, M_i, R_i\}$, $\mathcal{A}$ can impersonate U_i since then through the following method:

Step 1. Computes $R' = q_s \cdot R_i$.

Step 2. Replay the message $\{X_i, CID_i, M_i, R_i\}$ to S.

Step 3. Upon receiving the response $\{T_s, H_s\}$ from S, U_i computes $R_s^* = T_s - R'$, $H_{RS} = H(R' \parallel R_s^*)$ and $SK = H_3(r_i \cdot R_s^*)$.

Step 4. $U_i \rightarrow S : \{H_{RS}\}$.

After receiving the login request $\{X_i, CID_i, M_i, R_i\}$ sent by $\mathcal{A}$ in Step 2, S will perform Step $L3$ and $L4$ of the login and authentication phase (see Sec. II-C). It is easy to see that S will find no abnormality, because the login request is actually computed by the legitimate user U_i and indeed valid. Hence, S will proceed to compute R_s, T_s, H_s as usual and sends $\{T_s, H_s\}$ to U_i. In Step 3 stated above, the value of H_{RS} is indeed valid as $\mathcal{A}$ has computed the correct R' and $R_s^* (= T_s - R')$. As a result, upon receiving $\{H_{RS}\}$ sent by $\mathcal{A}$ in Step 4, S will find H_{RS}^* equal to the received H_{RS} in Step $L8$ of the verification phase. Therefore, server S will accept U_i's (actually $\mathcal{A}$'s) login request. In the end, server S and $\mathcal{A}$ will hold the same session key $SK = H_3(r_i \cdot R_s^*) = H_3(r_s \cdot R_i^*)$. By generalizing the above attack, $\mathcal{A}$ can easily imitate any user to login S at any time without employing any special cryptographic techniques. Hence, Truong et al.'s scheme cannot withstand KCI attack.

Remark 1. As revealed in [34], any authentication scheme in which the authentication server also serves as the registration center is inherently unable to achieve KCI resistance. A natural solution is to establish a new registration center for user registration (i.e., for the generation of user credentials), yet this may lead to the changes of user habits and thus downgrade of user experience. Fortunately, we find a more desirable solution. We are inspired by the recent proliferation of two-server password authentication schemes (see [35]) where the capability to verify user credentials are split up over two or more servers, so that KCI security still holds unless over a threshold of them are breached. In the meantime, users have no perception of protocol change. Our improvement proposed in Section VI is based on exactly this idea.

C. Poor usability

In Truong et al.'s scheme, a user needs to remember her identity ID_i and the authentication key AID_i generated by the server, where $AID_i = q_s \cdot H_1(ID_i \parallel X_i)$. As $H_1(\cdot)$ is a hash function, AID_i will be a *random* value but not meaningful phrase, and hence it is inconvenient for U_i to remember it. Therefore, Truong et al.'s scheme is not user friendly.

IV. Cryptanalysis of Li et al.'s scheme

In the above analysis, we have shown that it is really not an easy task to get a two-entity-involved (i.e., a user and a server) authentication scheme right; In the following, we will demonstrate that designing a three-entity-involved roaming authentication scheme can only be more challenging.

At GLOBECOM12, Li et al. [18] pointed out that He et al.'s roaming authentication scheme [17] is short of forward secrecy and key privacy and requires an expensive bilinear pairing operation on the user side. To overcome these defects, Li et al. [18] suggested a new ID-based scheme without using any bilinear pairing operations. Their scheme has two versions: one with user anonymity and the other not. In this work, we mainly focus on the one with user anonymity. Li et al. [18] claimed that their new scheme can preserve user privacy and withstand various known attacks, while achieving high efficiency due to the elimination of bilinear pairing operations. However, in this section we reveal that, essentially, Li et al.'s scheme achieves user anonymity at the cost of greatly reducing efficiency.

A. Review of Li et al.'s scheme

In this section, we briefly recall Li et al.'s scheme [18] and readers are referred to [18] for more details. This scheme involves three parties: the mobile user MU, a foreign server FS and the home server HS, and each server i holds a public-private key pair ($mpk_i = s_i \cdot P$, $msk_i = s_i$), where $s_i \in_R Z_p^*$. This scheme aims to provide the *strong anonymity* property: not only is the privacy of MU protected against the unauthorized external parties, but also against FS. That is, FS has no knowledge of the real identity of MU during the authentication process. The notations are listed in Table III.

TABLE III
NOTATIONS AND ABBREVIATIONS

Symbol	Description	Symbol	Description
MU	mobile user	ID_{HS}	identity of HS
HS	home server	ID_{FS}	identity of FS
FS	foreign server	mpk_i	public key of server i
ID_{MU}	identity of MU	msk_i	private key of server i

Setup phase. When a mobile user MU with identity ID_{MU} attempts to register with an authentication server (called home server HS), HS picks a family of pseudo-IDs $PID = \{pid_1, pid_2, \cdots\}$ where pid_i has a particular format and not been used before. For each pid_i, HS selects $r_i \in Z_p^*$ and calculates $R_{pid_i} = r_i \cdot P$ and $h_i = H(pid_i \| R_{pid_i})$. Then HS computes $s_{pid_i} = r_i + h_i \cdot msk_{HS}$ which matches pid_i, where msk_{HS} is HS's secret key. Finally, HS issues the pseudo-ID pid_i and the corresponding private key $\{R_{pid_i}, s_{pid_i}\}$ to MU via a secure channel. After receiving $\{R_{pid_i}, s_{pid_i}\}$. MU can verify whether $s_{pid_i}P = R_{pid_i} + H(pid_i \| R_{pid_i}) \cdot mpk_{HS}$ holds. If they are equal, it indicates $\{R_{pid_i}, s_{pid_i}\}$ is valid, otherwise it is invalid. Similarly, MU obtains a family of pseudo-IDs $PID = \{pid_1, pid_2, \cdots\}$ and private keys $\{(R_{pid_1}, s_{pid_1}), (R_{pid_2}, s_{pid_2}), ...\}$.

Roaming authentication phase. When MU visits the foreign server FS, MU selects an unused pseudo-ID pid_i as well as the corresponding private key (R_{pid_i}, s_{pid_i}). Then MU conducts the authentication procedure and agrees on a session key using pid_i and (R_{pid_i}, s_{pid_i}) with FS.

(1) MU picks $a \in_R Z_p^*$ and calculates aP and $K_1 = a \cdot mpk_{FS} \cdot P$. Then MU sends $\{aP, pid_i, R_{pid}, ID_{HS}\}$ to FS.

(2) After receiving $\{aP, pid_i, R_{pid}, ID_{HS}\}$, FS computes $K_1 = msk_{FS} \cdot aP$ by using its private key msk_{FS} and selects $b \in_R Z_p^*$ and calculates bP. Lets $K_2 = b \cdot (R_{pid_i} + H(pid_i \| R_{pid_i}) \cdot mpk_{HS})$ and $Auth_{FS} = MAC_{K_1}(aP, bP)$. FS sends $(bP, Auth_{FS})$ to MU.

(3) After receiving $(bP, Auth_{FS})$, MU validates whether $Auth_{FS} = MAC_{K_1}(aP, bP)$ holds. If they are equal, MU calculates $K_2 = msk_{MU} \cdot bP$ and $K_3 = a \cdot bP$. Let MU's authenticator $Auth_{MU} = MAC_{K_2}(aP)$ and the session key $sk_{MU\_FS} = H(pid_i \| ID_{FS} \| K_1 \| K_2 \| K_3)$. MU sends the authentication message $Auth_{MU}$ to FS.

(4) Upon receiving $Auth_{MU}$, foreign server FS validates whether $Auth_{MU} = MAC_{K_2}(aP)$ holds. If they are equal, FS calculates $K_3 = b \cdot aP$ and the session key $sk_{FS\_MU} = H(pid_i \| ID_{FS} \| K_l \| K_2 \| K_3) = sk_{MU\_FS}$.

B. An inherent design weakness

Although Li et al. does not illustrate the underlying rationale of the design of their protocol, the setup phase (see Sec. IV-A) is reminiscent of the famous BNN ID-based Signature (IBS) scheme proposed by Bellare et al. in [36]. Essentially, Li et al.'s scheme is based on BNN-IBS, which does not involve any bilinear pairing operations and thus is very suitable for mobile devices. However, achieving user anonymity only using BNN-IBS is still a challenging issue. Li et al. [18] made an attempt, yet we demonstrate that they once again fail.

In Li et al.'s scheme, the anonymity of a mobile user is attained by employing a pool of pseudo-IDs $PID = \{pid_1, pid_2, \cdots\}$ issued by the home server, and in each login a new pseudo-ID pid_j is used. We now emphasize that, in the roaming authentication phase, the foreign server FS has to check whether pid_j has already been used by MU before, otherwise a replay attack will definitely succeed. To check whether pid_j has already been used, FS has to maintain a timely updated list which stores all the used (expired) pseudo-IDs of every user. In other words, for n users in the system, every home/foreign authentication server shall effectively maintain (store) a revocation list of all pseudo-IDs that have been used by these n users. This further implies that, once a pseudo-ID has been used, it should be broadcasted to all the other servers in the system, otherwise this pseudo-ID along the corresponding cryptographic credentials can be replayed by attackers or malicious users. This greatly increases the management cost and communication overhead of the scheme. Actually, this issue is rather similar to the certificate revocation issue in PKI systems. Consequently, the efficiency of Li et al.'s scheme is far from satisfactory. As far as we know, there is no easy solution to this long-standing issue.

Remark 2. As being rigorously proved in [6], user anonymity can only be achieved by using some public-key primitives. Furthermore, while pre-loading a pseudo-IDs pool like Li et al.'s scheme or using Group/ring signatures do not seem feasible for light-weight mobile devices, public-key encryption schemes that are indistinguishable against adaptive chosen cipher-text attacks (IND-CCA2) along with a proper Hash padding mechanism would be promising candidates to solve the privacy-preserving issue [6]. Thus, only using BNN-IBS is not sufficient for achieving user privacy, and ID-based encryption (IBE) schemes shall be additionally used. Fortunately, a number of pairing-free IBE schemes (e.g., [12], [37]) have recently been developed, and they can be readily adopted to construct privacy-preserving authentication schemes which provide robust security and strong user anonymity.

V. Cryptanalysis of Zhang et al.'s scheme

In 2015, Zhang et al. [19] investigated into the requirements and issues in designing a secure and efficient authentication scheme for roaming services in mobile networks, and proposed a new scheme to eliminate the defects in existing schemes in [20], [21] by using ID-based cryptography. However, to the best of our knowledge, Zhang et al.'s work [19] as well as all the other previous literature in this area only pay attention to the threats arising from external adversaries *or* legitimate yet malicious foreign servers, *overlooking the challenges arising from the collusion of these two kinds of entities*. Here we use Zhang et al.'s scheme as a case study [19] and illustrate the effectiveness of this new kind of attack — collusion attack, in which an adversary colludes with a *legitimate yet curious* foreign server can effectively offline guessing MU's password. We also point out that several recent schemes (e.g., [22], [23]) are prone to this attack, which cannot be eliminated easily.

A. Review of Zhang et al.'s scheme

As with Li et al.'s scheme [18], Zhang et al.'s scheme [19] also deals with roaming authentication in mobile networks, and involves a mobile user (MU), a home server (HS) and a foreign server (FS). Notations are listed in Table I and III.

Setup phase. To gain ubiquitous service, a MU first shall register her to HS via a secure channel:

- Step R1. MU chooses her identity ID_{MU} and password PW_{MU}, and computes $V = h(PW_{MU} \| m)$, where $m \in_R \mathbb{Z}_p^*$.
- Step R2. $U_i \Rightarrow S : \{ID_i, V\}$.
- Step R3. HS chooses $n \in_R \mathbb{Z}_p^*$, and computes $MID = [ID_{MU} \| n]_K$ and $C = V \oplus h(ID_{MU} \| K)$, where K is HS's master key and $[\,]_K$ denotes symmetric encryption under key K.
- Step R4. $S \Rightarrow U_i$: A card with data $\{MID, C, ID_{HS}\}$.
- Step R5. MU stores m into the smart card.

Login phase. When a registered MU visits a foreign network charged by HS, MU and HS can authenticate each other and establish a session key as follows:

- Step L1. MU inserts her card into a card reader and inputs her password. The smart card selects $a \in_R \mathbb{Z}_p^*$, and computes $V = h(PW_{MU} \| m)$, $V' = V \oplus C$ and $Auth_{MU} = [ID_{MU} \| a \cdot P]_{V'}$
- Step L2. $MU \rightarrow FS : \{ID_{HS}, MID, Auth_{MU}\}$.
- Step L3. FS selects $b \in_R \mathbb{Z}_p^*$ and $D_{FS} = [ID_{FS} \| b \cdot P \| T_{FS}]_{K_{FH}}$, where T_{FS} is the current timestamp and K_{FH} is the symmetric secret key that is shared between FS and HS.
- Step L4. $FS \rightarrow HS : \{ID_{HS}, MID, Auth_{MU}, D_{FS}\}$.
- Step L5. HS decrypts D_{FS} to get $\{ID_{FS} \| b \cdot P \| T_{FS}\}$ and MID to get $\{ID_{MU} \| n\}$, and rejects if the decrypted ID_{MU} and T_{FS} are not valid.
- Step L6. HS computes $V' = h(ID_{MU} \| K)$ and decrypts $Auth_{MU}$ to get $\{ID_{MU} \| a \cdot P\}$.
- Step L7. HS selects $n' \in_R \mathbb{Z}_p^*$ and computes $MID' = [ID_{MU} \| n']_K$, $D_{HS} = [ID_{FS} \| b \cdot P \| T_{FS}]_{K_{FH}}$ and $Auth_{HS} = [MID' \| a \cdot P \| b \cdot P]_{V'}$.
- Step L8. $HS \rightarrow FS : \{D_{HS}, Auth_{HS}, T_{HS}\}$.
- Step L9. FS decrypts D_{HS} and checks the validity of T_{HS} and $b \cdot P$. If T_{HS} is within the allowed interval and $b \cdot P$ equals the value it computes in Step L3, then FS computes $Auth_{FS} = h(ab \cdot P \| ID_{MU} \| ID_{FS})$ and the session key $SK_{MF} = h(aP \| bP \| abP \| ID_{MU} \| ID_{FS})$.
- Step L10. $FS \rightarrow MU : \{Auth_{FS}, Auth_{HS}, ID_{FS}\}$.
- Step L11. MU decrypts $Auth_{HS}$ to get $\{MID' \| a \cdot P \| b \cdot P\}$ by using V', and rejects if the decrypted $b \cdot P$ does not equal the value it computes in Step L1.
- Step L12. MU computes $Auth^*_{FS} = h(ab \cdot P \| ID_{MU} \| ID_{FS})$ and rejects if the computed $Auth^*_{FS}$ does not equal the received $Auth_{FS}$.
- Step L13. MU computes the session key $SK_{MF} = h(aP \| bP \| abP \| ID_{MU} \| ID_{FS})$ and replaces the value of MID in her card memory with MID'.

B. Two security flaws in Zhang et al.'s scheme

Collusion attack. In 2015, Zhang et al. [19] pointed out that there are various defects in existing schemes (e.g., [20], [21]), and claimed that their new scheme "takes advantage of well-known schemes, achieving all security requirements of anonymous authentication while avoiding the weaknesses of current schemes." Besides, they also provided a formal security proof for their scheme under the intractability of Elliptic Curve Diffie-Hellman problem in the random oracle model.

Indeed, except for a minor defect that may lead to a replay attack (which, as will show later, can be easily addressed), Zhang et al.'s scheme [19] can withstand various known attacks that have been discussed in the literature. However, based on our past cryptanalysis experience on analyzing authentication schemes for wireless sensor networks where three entities are involved [6], we observe that collusion attacks may also be effective in mobile roaming authentication schemes. In such attacks, an external adversary $\mathcal{A}$ colludes with some legitimate yet curious insider (e.g., FS or MU, both of which can not be fully trusted) to attain goals that are beyond their respective capabilities. This kind of attack may lead to the breach of user anonymity and/or disclosure of user password, which is quite damaging. It has been extensively discussed in user authentication schemes for wireless sensor networks [6], yet as far as we know, little attention has been given to it in the research area of roaming authentication.

Now let's see how this attack could be effectively launched with Zhang et al.'s roaming authentication scheme in place. Suppose MU's smart card has been stolen or picked up by an adversary $\mathcal{A}$, and the sensitive data $\{V, m\}$ in the card memory can be extracted by using side-channel attacks or reverse engineering [38], [39]. With the previously eavesdropped protocol transcript $\{Auth_{MU}\}$ that were exchanged among MU, FS and HS, $\mathcal{A}$ can obtain MU's password PW_{MU} with the help of a legitimate yet curious FS as follows:

Step 1. Guesses the value of PW_{MU} to be PW^*_{MU} from a password dictionary $\mathcal{D}_{pw}$ and the value of ID_{MU} to be ID^*_{MU} from an identity dictionary $\mathcal{D}_{id}$;

Step 2. Computes $V^* = V \oplus H_1(PW^*_{MU} \| m)$, where V and m are extracted from MU's smart card;

Step 3. Computes $Auth^*_{MU} = [ID^*_{MU} \| a \cdot P]_{V^*}$, where $a \cdot P$ is obtained with the help of the curious FS;

Step 4. Verifies the correctness of (ID^*_{MU}, PW^*_{MU}) by checking if the computed $Auth^*_{MU}$ equals the intercepted $Auth_{MU}$.

Step 5. Repeats Steps 1, 2, 3 and 4 of this procedure until the correct pair of $(ID_{MU},\ PW_{MU})$ is found.

The time complexity of the above attack procedure is $\mathcal{O}(|\mathcal{D}_{id}| * |\mathcal{D}_{pw}| * (T_S + T_H))$, where $|\mathcal{D}_{id}|$ and $|\mathcal{D}_{pw}|$ denote the size of the identity space and password space, respectively; T_S is the running time for symmetric encryption and T_H is the running time for Hash function. Generally, the password space and identity space are very limited in practice, e.g., $|\mathcal{D}_{id}| \leq |\mathcal{D}_{pw}| \leq 10^6$ [26], [27]. According to the timings in Table II, MU's password can be offline guessed in about 21.75 hours on a common Intel i5-2450M 2.50 GHz processor.

It is crucial to note that, MU's authenticator $V' = h(ID_{MU} \| K)$ is concealed (encrypted) in $Auth_{MU} = [ID_{MU} \| a \cdot P]_{V'}$ by only using a symmetric encryption. As a result, there is no randomness involved in this transformation. This means that if $\mathcal{A}$ obtains ID_{MU} and $a \cdot P$, she can *definitely* determine V' by guessing PW_{MU}. On the other hand, some randomness would be introduced if V' is concealed by using some IND-CCA2 secure public-key encryption algorithm (e.g., [12], [37]) like the schemes in [30], [37], and this may inevitably lose some efficiency. Halevi and Krawczyk [11] have confirmed that public-key techniques are indispensable for password-based protocols to resist offline guessing attacks. This indicates that a number of recent schemes (e.g., [22], [23]) that only use symmetric-key techniques to conceal a user's authenticator are inherently prone to this flaw. Hence, all the schemes in [19], [22], [23] cannot be easily remedied and have to employ some public-key techniques to eliminate the identified security flaw.

Replay attack. Besides, this scheme is also susceptible to replay attack. It is easy to see that, in MU's login request $\{ID_{HS}, MID, Auth_{MU}\}$ there is no mechanism for HS to verify the freshness of the data. Any previously legitimate login request can be replayed by $\mathcal{A}$ to impersonate as MU, and HS can not detect this malicious behavior and will respond to MU (actually, $\mathcal{A}$) as usual. Thankfully, since $\mathcal{A}$ does not know V', she cannot compute the session key $SK_{MF} = h(aP \| bP \| abP \| ID_{MU} \| ID_{FS})$. Still, $\mathcal{A}$ manages to make both HS and FS perform useless computations and communications and to make HS believe that MU is logining. In this light, it is quite undesirable. Fortunately, this attack can be easily eliminated by adding a timestamp to the login request. More specifically, MU now computes $Auth_{MU} = [ID_{MU} \| a \cdot P \| T_{MU}]_{V'}$ and sends $\{ID_{HS}, MID, T_{MU}, Auth_{MU}\}$ to FS, where T_{MU} is MU's current timestamp.

VI. An Improvement over Truong et al.'s Scheme

This section shall investigate into the rationales underlying the failures in Truong et al.'s scheme [16] and brief the corresponding countermeasures. The resulting protocol is illustrated in Fig.2, where all the changes have been underlined by dashed lines. As our improvement maintains all the merits while eliminating the identified pitfalls, it is more secure and user-friendly, and thus it is promising for practical applications.

Known session-specific information attack: The analysis in Sec. III-A has shown that, once U_i's ephemeral exponent r_i is leaked, $\mathcal{A}$ can figure out U_i's real identity ID_i, thereby breaching user anonymity. The core crux lies that, with r_i in hand, $\mathcal{A}$ now can repeatedly verify whether her guess ID^*_i is right or not by checking $R_i \overset{?}{=} r_i \cdot H_1(ID^*_i \parallel X_i)$, where R_i and X_i are obtained from the public channel. We note that, only in the login request can R_i be exploited by $\mathcal{A}$ to use as a comparison target for identity guessing, while other transcripts in $\{CID_i, M_i, T_s, H_s, H_{RS}\}$ cannot be exploited, for the mere reason that R_i is computed without the contribution of the long-term secret AID_i. Now, the countermeasure is obvious: computing R_i=$r_i \cdot H_1(ID^*_i \| X_i \| AID_i)$. In this way, known session-specific information attack would not be successful.

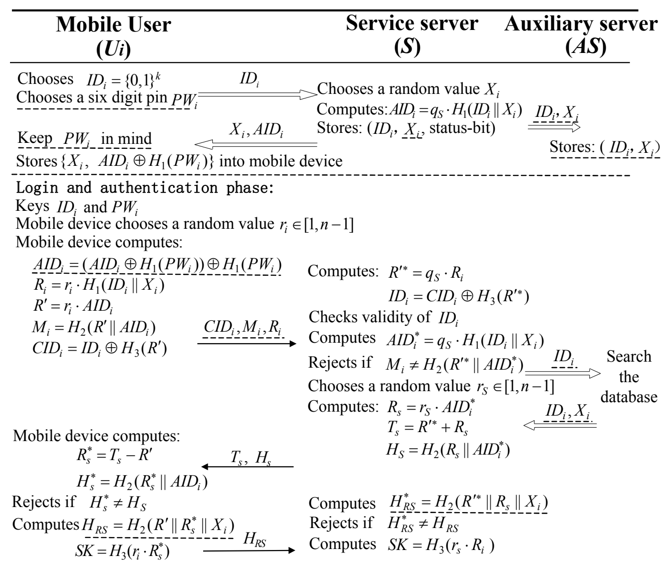

Fig. 2. An improved ID-based authentication scheme for mobile devices

An alternative (and more desirable) solution is to set up an auxiliary server AS (see Remark 1) that stores the parameter X_i. This means S and AS now keep an entry $\{ID_i,\ \texttt{status-bit}\}$ and $\{ID_i, X_i\}$ corresponding to U_i, respectively, and U_i only sends $\{CID_i, M_i,\ R_i\}$ as her login request. Without X_i, the attack in Sec. III-A will definitely fail. Particularly, in this way, the KCI attack in Sec. III-B can also be eliminated. So, that kills two birds with one stone.

Key compromise impersonation attack: The attack described in Sec. III-B succeeds due to the inherent reason that, the authentication server S also serves as the registration server. Thus, with the long-time secret key q_s of server S, $\mathcal{A}$ can first compute $R' = q_s \cdot R_i = q_s \cdot r_i \cdot H_1(ID_i \parallel X_i)$, and then compute U_i's authenticator $H_{RS} = H(R' \parallel (T_s - R'))$, where T_s can be intercepted from the public channel. However, as detailed in Remark 1, if we separately store X_i in the new auxiliary server AS and U_i only sends $\{CID_i, M_i, R_i\}$ as her login request and U_i's authenticator H_{RS} is computed as $H_{RS} = H(R' \parallel R_s \parallel X_i)$. Then, without either q_s or X_i, it is virtually impossible for $\mathcal{A}$ to carry out KCI attack, while q_s and X_i are protected by two different, distinct security architecture guarded and maybe physically located servers.

Usability problem: While users are incapable of memorizing a random value like AID_i (see the definition in Sec. II-B), they can instead remember a short string like a six-digit PIN denoted by PW_i. Accordingly, $AID_i \oplus PW_i$ is now stored in the mobile device, and whenever U_i logins S, she keys ID_i and PW_i (instead of ID_i and AID_i), and the device retrieves $AID_i = (AID_i \oplus PW_i) \oplus PW_i$.

VII. Conclusion

Understanding security and efficiency failures of cryptographic protocols is the key to both patching existing protocols and designing future schemes. In this work, we have shown that though Truong et al.'s scheme, Li et al.'s scheme and Zhang et al.'s scheme are very efficient and possess many attractive features, they, in fact, are unable to achieve some of the claimed important design goals. As all three schemes are improvements over existing schemes, our results suggest that simply amending a protocol for resistance against known attacks, yet paying little attention to the roots of the identified failures, does not always yield a more robust one.

Particularly, our cryptanalysis results on Zhang et al.'s "provably secure" scheme (and the recent schemes in [22], [23]) highlight that providing a "formal proof" is no panacea for assuring security and it is of great importance to be aware of potential threats when designing a protocol. This suggests the necessity of this work. We further investigate into *the rationales of the identified failures* and put forward viable fixes for Truong et al.'s scheme, while we find there is no simple countermeasure to the issues of Li et al.'s and Zhang et al.'s schemes. Since our improvement incurs reasonable cost while fixing the security loopholes and providing better usability, it is more promising for practical applications.

References

[1] R. Mahindra, H. Viswanathan, K. Sundaresan, M. Y. Arslan, and S. Rangarajan, "A practical traffic management system for integrated lte-wifi networks," in *Proc. MobiCom 2014*, 2014, pp. 189–200.

[2] D. He and S. Zeadally, "Authentication protocol for an ambient assisted living system," *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 71–77, 2015.

[3] K. Zhang, K. Yang, X. Liang, Z. Su, and X. S. Shen, "Security and privacy for mobile healthcare networks: from a quality of protection perspective," *IEEE Wirel. Commun.*, vol. 22, no. 4, pp. 104–112, 2015.

[4] M. Guizani, D. He, K. Ren, J. Rodrigues, S. Chan, and Y. Zhang, "Security and privacy in emerging networks," *IEEE Commun. Mag.*, 2015, doi:10.1109/MCOM.2015.7081098.

[5] J.-L. Tsai and N.-W. Lo, "A privacy-aware authentication scheme for distributed mobile cloud computing services," *IEEE Syst. J.*, vol. 9, no. 3, pp. 805–815, 2015.

[6] D. Wang and P. Wang, "On the anonymity of two-factor authentication schemes for wireless sensor networks: Attacks, principle and solutions," *Comput. Netw.*, vol. 73, pp. 41–57, 2014.

[7] B. Neuman and T. Ts'o, "Kerberos: An authentication service for computer networks," *IEEE Commun. Mag.*, vol. 32, no. 9, pp. 33–38, 1994.

[8] H. Krawczyk, "HMQV: A high-performance secure diffie-hellman protocol," in *Proc. CRYPTO 2005*. Springer, 2005, pp. 546–566.

[9] B. LaMacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in *Proc. ProvSec 2007*, ser. LNCS, W. Susilo, J. Liu, and Y. Mu, Eds. Springer, vol. 4784, pp. 1–16.

[10] "Public-key cryptography standards, PKCS#11 mechanisms v2.30," Dec., 2014, http://www.rsasecurity.com/rsalabs/pkcs/.

[11] S. Halevi and H. Krawczyk, "Public-key cryptography and password protocols," *ACM Trans. Inform. Syst. Secur.*, vol. 2, pp. 230–268, 1999.

[12] G. Yang and C. H. Tan, "Certificateless public key encryption: A new generic construction and two pairing-free schemes," *Theor. Comput. Sci.*, vol. 412, no. 8, pp. 662–674, 2011.

[13] K. Lauter, "The advantages of elliptic curve cryptography for wireless security," *IEEE Wirel. Commun.*, vol. 11, no. 1, pp. 62–67, 2004.

[14] J. Yang and C. Chang, "An id-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Comput. Secur.*, vol. 28, no. 3-4, pp. 138–143, 2009.

[15] S. H. Islam and G. Biswas, "A more efficient and secure id-based remote mutual authentication with key agreement scheme for mobile devices on ecc," *J. Syst. Soft.*, vol. 84, no. 11, pp. 1892 –1898, 2011.

[16] T. T. Truong, M. T. Tran, and A. D. Duong, "Improvement of the more efficient and secure id-based remote mutual authentication with key agreement scheme for mobile devices on ecc," in *Proc. AINA 2012*. IEEE, 2012, pp. 698–703.

[17] D. He, C. Chen, S. Chan, and J. Bu, "Secure and efficient handover authentication based on bilinear pairing functions," *IEEE Trans. Wirel. Commun.*, vol. 11, no. 1, pp. 48–53, 2012.

[18] X. Li, Y. Zhang, X. Liu, J. Cao, and Q. Zhao, "A lightweight roaming authentication protocol for anonymous wireless communication," in *Proc. GLOBECOM 2012*. IEEE, 2012, pp. 1029–1034.

[19] G. Zhang, D. Fan, Y. Zhang, X. Li, and X. Liu, "A privacy preserving authentication scheme for roaming services in global mobility networks," *Secur. Commun. Netw.*, 2015, doi:10.1002/sec.1209.

[20] C. Chang and H. Tsai, "An anonymous and self-verified mobile authentication with key agreement for large-scale wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 9, no. 11, pp. 3346–3353, 2010.

[21] W.-C. Kuo, H.-J. Wei, and J.-C. Cheng, "An efficient and secure anonymous mobility network authentication scheme," *J. Inform. Secur. Appl.*, vol. 19, no. 1, pp. 18–24, 2014.

[22] P. Gope and T. Hwang, "Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks," *IEEE Syst. J.*, 2015, doi:10.1109/JSYST.2015.2416396.

[23] M. Farash, S. Chaudhry, and et al., "A lightweight anonymous authentication scheme for consumer roaming in ubiquitous networks with provable security," *Int. J. Commun. Syst.*, 2015, doi: 10.1002/dac.3019.

[24] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proc. EUROCRYPT 2001*, ser. LNCS, B. Pfitzmann, Ed. Springer, 2001, vol. 2045, pp. 453–474.

[25] 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.

[26] J. Bonneau, "The science of guessing: Analyzing an anonymized corpus of 70 million passwords," in *Proc. IEEE S&P 2012*, 2012, pp. 538–552.

[27] D. Wang and P. Wang, "The emperor's new password creation policies," in *Proc. ESORICS 2015*, ser. LNCS. Springer, vol. 9327, pp. 1–22.

[28] *Miracl library*, Shamus Software Ltd., May 2013, http://www.shamus.ie/index.php?page=home.

[29] *Standards for Efficient Cryptography, SEC 2: Recommended Elliptic Curve Domain Parameters, Version 2.0*, Certicom Research, Jan. 2010, available at http://www.secg.org/download/aid-784/sec2-v2.pdf.

[30] H. Jo, J. Paik, and D. Lee, "Efficient privacy-preserving authentication in wireless mobile networks," *IEEE Trans. on Mob. Comput.*, vol. 13, no. 7, pp. 1469–1481, 2014.

[31] D. He, S. Chan, and M. Guizani, "Handover authentication for mobile networks: security and efficiency aspects," *IEEE Netw.*, vol. 29, no. 3, pp. 96–103, 2015.

[32] "50 million compromised in evernote hack," Mar. 2013, http://www.cnn.com/2013/03/04/tech/web/evernote-hacked/.

[33] "Heartbleed – openssl zero-day bug leaves millions of websites vulnerable," April 2014, http://www.tuicool.com/articles/yyUfUz.

[34] D. Wang, N. Wang, P. Wang, and S. Qing, "Preserving privacy for free: Efficient and provably secure two-factor authentication scheme with user anonymity," *Info. Sci.*, vol. 321, pp. 162–178, 2015.

[35] J. Camenisch, R. R. Enderlein, and G. Neven, "Two-server password-authenticated secret sharing uc-secure against transient corruptions," in *Proc. PKC 2015*, pp. 283–307.

[36] M. Bellare, C. Namprempre, and G. Neven, "Security proofs for identity-based identification and signature schemes," *J. Cryptol.*, vol. 22, no. 1, pp. 1–61, 2009.

[37] R. Yasmin, E. Ritter, and G. Wang, "Provable security of a pairing-free one-pass authenticated key establishment protocol for wireless sensor networks," *Int. J. Inform. Secur.*, vol. 13, no. 5, pp. 453–465, 2014.

[38] K. Nohl, D. Evans, S. Starbug, and H. Plötz, "Reverse-engineering a cryptographic RFID tag," in *Proc. USENIX SEC 2008*, pp. 185–193.

[39] J. Liu, Y. Yu, and F.-X. Standaert, "Small tweaks do not help: Differential power analysis of milenage implementations in 3G/4G USIM cards," in *Proc. ESORICS 2015*. Springer, 2015, pp. 1–20.