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 acquisition of corroborative evidence.

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 pair 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-unfriendly.

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 \( G_p \).
B. The registration phase

The registration phase involves the following operations:

Step L1. $U_i$ chooses her identity $ID_i \in_R \{0, 1\}^k$.

Step L2. $U_i \rightarrow S : \{ID_i\}$.

Step L3. Upon receiving the login request from $U_i$, $S$ computes $R_i^s := q_s \cdot R_i$, $ID_i^s := CID_i \parallel H_2(R_i^s)$, and then checks the validity of the identity $ID_i^s$. If $ID_i^s$ is valid, $S$ goes to the next step, otherwise rejects the login request.

Step L4. $S$ computes the authentication key $AIID_i^s := q_s \cdot H_1(ID_i^s \parallel X_i)$ and checks whether $H_2(R_i^s \parallel AIID_i^s)$ 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 $R_s = r_s \cdot AIID_i^s$, $T_s = R_i^s + R_s$ and $H_s = H_2(R_s \parallel AIID_i^s)$.

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

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

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

Step L8. The server $S$ computes $H_{RS} = H_2(R_i^r \parallel RS)$ 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 $SK = H_3(r_s \cdot RS)$, otherwise rejects.

III. CRYPTOANALYSIS 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 $A$. Once the login request message $\{X_i, CID_i, M_i, R_i\}$ during any authentication process is intercepted by $A$, $A$ can obtain $U_i$’s identity as follows:

Step 1. Guesses the value of $ID_i$ to be $ID_i^* \in_R D_{id}$.

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

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

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 $|D_{id}|$ denote the size of the identity dictionary $D_{id}$. The time
complexity of the above attack procedure is \( O(\|D_{id}\| \cdot (T_P + T_H)) \), 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 \( A \) to recover \( U_i \)’s identity is a linear function of the identity dictionary size \( |D_{id}| \).

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 \( D_{id} \) shall be very limited in practice, e.g., \( \|D_{id}\| \leq |D_{pw}| \leq 10^6 \) [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 \( \mu \)s 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

<table>
<thead>
<tr>
<th>Computation Evaluation of Related Operations on Laptop PCs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Experimental Platform (common PCs)</strong></td>
</tr>
<tr>
<td>Intel i5-5200 2.30 GHz</td>
</tr>
<tr>
<td>Intel i5-2210 2.30 GHz</td>
</tr>
<tr>
<td>Intel i5-2450M 2.50 GHz</td>
</tr>
</tbody>
</table>

Once the correct value of \( ID_i \) is obtained by \( A \), 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 \( A \) is assumed to be with the knowledge of a communicating party \( i \)’s private key, and this enables \( A \) not only to masquerade party \( i \) to others but also to masquerade other uncompromised 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 access 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 \( A \) 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 \( A \) to impersonate \( S \), \( A \) 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, \( A \) 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 \( A \). Without loss of generality, we suppose one of \( U_i \)’s previous login requests, say \( \{X_i, CID_{i1}, M_i, R_i\} \), is intercepted by \( A \). Once the value of \( q_s \) is obtained, with the previously intercepted protocol transcripts \( \{X_i, CID_{i1}, M_i, R_i\} \), \( A \) can impersonate \( U_i \) since then through the following method:

1. **Step 1.** Computes \( R' = q_s \cdot R_i \).
2. **Step 2.** Replays the message \( \{X_i, CID_{i1}, M_i, R_i\} \) to \( S \).
3. **Step 3.** Upon receiving the response \( \{T_s, H_s\} \) from \( S \), \( U_i \) computes \( R'_s = T_s - R' \), \( H_{RS} = H(R' \| R'_s) \) and \( SK = H_3(r_i \cdot R'_s) \).
4. **Step 4.** \( U_i \rightarrow S: \{H_{RS}\} \).

After receiving the login request \( \{X_i, CID_{i1}, M_i, R_i\} \) sent by \( A \) 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 \( A \) has computed the correct \( R' \) and \( R'_s = (T_s - R') \). As a result, upon receiving \( \{H_{RS}\} \) sent by \( A \) 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 \( A \)’s) login request. In the end, server \( S \) and \( A \) will hold the same session key \( SK = H_3(r_i \cdot R'_s) = H_3(r_s \cdot R'_s) \). By generalizing the above attack, \( A \) 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 \( AID_i = q_s \cdot H_1(ID_i || X_i) \). 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 \((mpk_i, sk_i)\), where \(s_i \in \mathbb{Z}_p^*\). 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>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MU</td>
<td>mobile user</td>
</tr>
<tr>
<td>HS</td>
<td>home server</td>
</tr>
<tr>
<td>FS</td>
<td>foreign server</td>
</tr>
<tr>
<td>ID(_{MU})</td>
<td>identity of MU</td>
</tr>
<tr>
<td>ID(_{HS})</td>
<td>identity of HS</td>
</tr>
<tr>
<td>ID(_{FS})</td>
<td>identity of FS</td>
</tr>
<tr>
<td>mpk(_{i})</td>
<td>public key of server (i)</td>
</tr>
<tr>
<td>sk(_{i})</td>
<td>private key of server (i)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>a (\in \mathbb{Z}<em>p^*) and calculates (aP) and (K_1 = a \cdot mpk</em>{FS} \cdot P). Then MU sends ({aP, pid_i, R_{pid_i}, ID_{HS}}) to FS.</td>
</tr>
<tr>
<td>b</td>
<td>Upon receiving ({aP, pid_i, R_{pid_i}, ID_{HS}}), FS computes (K_1 = msk_{FS} \cdot aP) by using its private key (msk_{FS}) and selects (b \in \mathbb{Z}<em>p^*) and calculates (bP). Let (K_2 = b \cdot (R</em>{pid_i} + H(pid_i</td>
</tr>
<tr>
<td>c</td>
<td>Upon receiving ((bP, Auth_{FS})), MU validates whether (Auth_{FS} = MAC_{K_2}(aP, bP)) holds. If they are equal, MU computes (K_2 = msk_{MU} \cdot bP) and (K_3 = a \cdot bP). Let MU’s authenticator (Auth_{MU} = MAC_{K_2}(aP, bP)) and the session key (sk_{MU} = H(pid_i</td>
</tr>
<tr>
<td>d</td>
<td>Upon receiving (Auth_{MU}), foreign server FS validates whether (Auth_{MU} = MAC_{K_2}(aP, bP)) holds, if they are equal, FS calculates (K_3 = b \cdot aP) and the session key (sk_{FS-MU} = H(pid_i</td>
</tr>
</tbody>
</table>

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 \(PID = \{pid_1, pid_2, \ldots\}\) issued by the home server, and in each login a new pseudo-ID \(pid_i\) is used. We now emphasize that, in the roaming authentication phase, the foreign server FS has to check whether \(pid_i\) has already been used by MU before, otherwise a replay attack will definitely succeed. To check whether \(pid_i\) 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 L1. MU chooses her identity $ID_{MU}$ and password $PW_{MU}$, and computes $V = h(PW_{MU}|m)$, where $m \in R Z_p^*$.

Step L2. $U_i \Rightarrow S : \{ID_i, V\}$.

Step L3. HS chooses $n \in R Z_p^*$, and computes $MID = [ID_{MU}|m]_K$ and $C = V \oplus h(ID_{MU}|m)_K$, where $K$ is HS’s master key and $[.]_K$ denotes symmetric encryption under key $K$.

Step L4. $S \Rightarrow U_i$: A card with data $\{MID, C, ID_{HS}\}$.

Step L5. MU stores $m$ into the smart card.

Login phase. When a registered MU visits a foreign network charged by MU, HS and MU 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 $a \in R Z_p^*$, and computes $V = h(PW_{MU}|m)$, $V' = V \oplus C$ and $Auth_{MU} = [ID_{MU}|a \cdot P]_K$.

Step L2. $MU \rightarrow FS : \{ID_{HS}, MID, Auth_{MU}\}$.

Step L3. $FS$ selects $b \in R Z_p^*$ and $D_{FS} = [ID_{FS}|b \cdot P]_{K_{FS}}$, where $T_{FS}$ is the current timestamp and $K_{FS}$ 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 $\{ID_{FS}|b \cdot P\}_{K_{FS}}$ and MID to get $\{ID_{MU}|n\}$, and rejects if the decrypted $ID_{MU}$ and $T_{FS}$ are not valid.

Step L6. HS computes $V' = h(ID_{MU}|n)|K$ and decrypts $Auth_{MU}$ to get $\{ID_{MU}|a \cdot P\}$.

Step L7. HS selects $n' \in R Z_p^*$ and computes $MID' = [ID_{MU}|n'|K_{FS}]_{K_{FS}}$, and rejects if $MID$ not equal the value it computes in Step L3.

Step L8. HS selects $n' \in R Z_p^*$ and computes $MID' = [ID_{FS}|b \cdot P|T_{FS}]_{K_{FS}}$, and rejects if $MID$ not equal the value $K_{FS}$.

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 L9, then FS computes $Auth_{FS} = h(a \cdot P|ID_{MU}|ID_{FS})$ and the session key $SK_{MF} = h(a|b|P|ab|P|ID_{MU}|ID_{FS})$.

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 accepts if the decrypted $b \cdot P$ does not equal the value it computes in Step L1.

Step L12. $MU$ computes $Auth_{FS}' = h(a \cdot P|ID_{MU}|ID_{FS})$ and rejects if the computed $Auth_{FS}'$ does not equal the received $Auth_{FS}$.

Step L13. $MU$ computes the session key $SK_{MF} = h(a|b|P|ID_{MU}|ID_{FS})$ 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 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 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 $HS$, 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 $D_{pw}$ and the value of $ID_{MU}$ to be $ID^{*}_{MU}$ from an identity dictionary $D_{id}$.

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 $O([D_{id}] \cdot |D_{pw}| \cdot (T_{S} + T_{H}))$, where $|D_{id}|$ and $|D_{pw}|$ 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., $|D_{id}| \leq |D_{pw}| \leq 10^{6}$ [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 $A$ 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. Haveli 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 $A$ to impersonate as $MU$, and $HS$ can not detect this malicious behavior and will respond to $MU$ (actually, $A$) as usual. Thankfully, since $A$ does not know $V^{*}$, she cannot compute the session key $SK_{MF} = h(aP||bP||ID_{MU}||ID_{FS})$. Still, $A$ manages to make both $HS$ and $FS$ perform useless communications and computations and to make $HS$ believe that $MU$ is logging in. 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, $A$ can figure out $U_{i}$’s real identity $ID_{i}$, thereby breaching user anonymity. The core crux lies that, with $r_{i}$ in hand, $A$ now can repeatedly verify whether her guess $ID_{i}^{*}$ is right or not by checking $R_{i} = r_{i} \cdot H_{1}(ID_{i}^{*} || X_{i})$, 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 $A$ to use as a comparison target for identity guessing, while other transcripts in \{CID, $M_{i}, T_{i}, H_{i}, H_{RS}$\} 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](image-url)
is virtually impossible for $A$ 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 $U_i$ logs in $S$, 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**


