Enhanced Security and Pairing-free Handover Authentication Scheme for Mobile Wireless Networks

Rui Chen\textsuperscript{1,2,a}, Guangqiang Shu\textsuperscript{2,b}, Peng Chen\textsuperscript{1,c} and Lijun Zhang\textsuperscript{3,d}

\textsuperscript{1}College of Computer Science, Sichuan University, Chengdu, China
\textsuperscript{2}College of Computer Science, Sichuan Normal University, Chengdu, China
\textsuperscript{3}Chengdu Ruibei Yingtong Information Technology Limited Company, Chengdu, China
\textsuperscript{a}crs1934@hotmail.com, \textsuperscript{b}254625466@qq.com, \textsuperscript{c}7199611@qq.com, \textsuperscript{d}630452696@qq.com

Abstract. With the widely deployment of mobile wireless networks, we aim to propose a secure and seamless handover authentication scheme that allows users to roam freely in wireless networks without worrying about security and privacy issues. Given the open characteristic of wireless networks, safety and efficiency should be considered seriously. Several previous protocols are designed based on a bilinear pairing mapping, which is time-consuming and inefficient work, as well as unsuitable for practical situations. To address these issues, we designed a new pairing-free handover authentication scheme for mobile wireless networks. This scheme is an effective improvement of the protocol by Xu et al., which is suffer from the mobile node impersonation attack. Security analysis and simulation experiment indicate that the proposed protocol has many excellent security properties when compared with other recent similar handover schemes, such as mutual authentication and resistance to known network threats, as well as requiring lower computation and communication cost.

1. Introduction

Mobile intelligent terminal technology and wireless communication with mobile wireless network (MWN) are currently being developed and used in several areas along with the rapid development of wireless networks. All types of mobile applications and mobile internet services in intelligent mobile nodes (MNs) (i.e., laptops, smartphones, tablets, and PDA) enhance the variation and convenience of our lives. However, network and information security are increasingly gaining attention with the development of various network attack technologies.

An anonymous mobile node handover authentication (AMNHA) protocol allows MNs to roam seamlessly over multiple access points (APs) in MWNs while providing mutual authentication for two sides of communication and protecting the privacy of MNs. Typical MWNs often comprise three types of entities, including MNs, APs, and an authentication server (AS)[1]. After registering to an AS, the MN can access the network and acquire network service from an associated AP. When an MN moves from one AP to another, it should perform a handover authentication to the new AP to protect them from unauthorized user access or unauthorized access by legitimate users. Meanwhile, a shared session key will be generated after the AMNHA protocol is executed to ensure communication data consistency, integrity, and accuracy.

Several protocols have been recently proposed by employing various methods. However, a secure and efficient roaming AMNHA protocol is difficult to design given the broadcast nature of wireless networks and the limited power and resources of mobile devices.
Based on bilinear pairings and identity-based signature, He et al.[2] presented a new AMNHA protocol, which uses a pseudo-ID to provide mutual authentication and privacy protection between communication parties. However, this protocol is insecure and prone to key compromised attack. Subsequently, a number of improved approaches have been proposed [3, 6, 7, 8]. He et al.[4] and He et al.[5] outlined some characteristics that a secure and efficient AMNHA protocol should satisfy and proposed different solutions based on bilinear pairings and elliptic curve cryptosystem (ECC).

However, these protocols [2-8] are unsuitable for use in a mobile environment, where a number of wireless mobile devices with limited power and processing capability are deployed, due to complexities and time-consuming operation in model cryptography that results from adopting a bilinear pairing operation [16]. To address the above mentioned problems, a number of efficient pairing-free methods and its corresponding protocols have been proposed [9-15]. In these protocols, the bilinear pairing operation is completely removed.

Cao et al.[9] presented a new identity-based AMNHA protocol without bilinear pairing. Their scheme can support strong security while minimizing the message exchange time. However, the scheme in [9] is not secure and vulnerable to two types of attacks as pointed out by [10]. Thereafter, Cao et al.[11] presented a new uniform handover authentication scheme based on their previous method[9]. Their improved approach provides strong security and robust efficiency while supporting user mutual authentication.

However, Li et al.[12] noted that the improved scheme of Cao et al. cannot provide user anonymity and user untraceability given that the real identity of the MN is transmitted in clear text form over the wireless networks. They then presented a new privacy-aware AMNHA protocol for MWNs, which is proven secure under an extended Canetti–Krawczyk model. By adopting the identity-based cryptography and pseudo-ID for the MN, their protocol can achieve user anonymity and good performance.

Unfortunately, Chaudhry et al.[13] and Xie et al.[14] indicated that Li et al.’s protocol still has a number of drawbacks, and an adversary may impersonate a legal AP to communicate with an innocent MN. Thus, the protocol is prone to potential fraud attacks and cannot provide mutual authentication between the MN and AP. They then proposed different solutions.

Xu et al.[15] recently presented a novel privacy preserving AMNHA scheme without pairing operation for MWNs. They claimed that their scheme has excellent security and efficiency while supporting user anonymity and untraceability. However, we found that this scheme remains vulnerable to MN impersonation attack, which will result in the failure of mutual authentication. Furthermore, we propose a new pairing-free AMNHA protocol that overcomes the weaknesses of Xu et al.’s protocol.

2. Review of Xu Yang et al.’s Scheme
This section gives a brief review of Xu et al.’s scheme [15]. Their protocol is divided into following two phases:

2.1 Pre-deployment Phase
The pre-deployment phase includes the following two processes:

1) System initialization
In this process, the AS needs to complete a few steps:

a) Let \( q \) be a \( k \)-bit prime. Selects a tuple \( \{ F_q, E/F_q, G, P \} \), where \( F_q \) is a finite field, \( E/F_q \) is an elliptic curve \( E \) over \( F_q \), \( G \) is a cyclic additive group of order \( q \), and \( P \) is a base point of \( G \) over \( E/F_q \).

b) Randomly selects two integers \( s, r \in \mathbb{Z}_q^* \) as the system master keys. Then, calculates \( PK_1 = s \cdot P \), \( PK_2 = r \cdot P \) is the corresponding public keys.

c) Selects four cryptographic hash functions \( H_1, H_2, H_3, \) and \( H_4 \), where \( H_1 : \{ 0, 1 \}^k \times G \rightarrow \mathbb{Z}_q^* \), \( H_2 : \{ 0, 1 \} \times \{ 0, 1 \}^k \rightarrow \{ 0, 1 \}^k \), \( H_3 : \{ 0, 1 \} \times \{ 0, 1 \}^k \rightarrow \{ 0, 1 \}^k \), \( H_4 : \{ 0, 1 \}^k \times \{ 0, 1 \} \rightarrow \{ 0, 1 \}^k \).

d) Publishes the public parameters (i.e., \( F_q, E/F_q, G, P, PK_1, PK_2, H_1, H_2, H_3, H_4 \)) and keeps \( s, r \) secret.

2) Key pre-distribution
a) For an MN, the AS selects a set of independent pseudo-IDs \((PID = pid_1, pid_2, \ldots)\) to replace the MN’s real identity \((i.e. ID_{MN})\). Then, the AS generates the private key \((S_{pid} = r + H_i(pid_i \| PK_2)s)\) for each pseudo-ID \((pid_i \in PID)\) and transfers all tuples \((pid_i, S_{pid})\) to the MN via a secure tunnel. The MN can calculate the corresponding public key \((PK_{pid} = S_{pid}P = PK_2 + H_i(pid_i \| PK_2)PK_i)\).

b) For an AP with real identity \(ID_{AP}\), the AS computes its private key \((S_{AP} = r + H_i(ID_{AP} \| PK_2)s)\) and the AP can then calculate the corresponding public key \((PK_{AP} = S_{AP}P = PK_2 + H_i(ID_{AP} \| PK_2)PK_i)\). At last the AS securely transfers the private key \(S_{AP}\) to the AP.

### 2.2 Handover Authentication Phase

The MN should perform the handover authentication process to access the network service when it leaves the current AP and moves into a new AP. The details are as follows.

1) \(MN \rightarrow AP: \{Mi, RMN, H_3(PTK_i \| RAP)\}\)

The MN picks a random integer \(a \in \mathbb{Z}^*_q\), generates a timestamp \(ts\), and computes \(R_{MN} = a \cdot P\). Then, the MN selects an unused pseudo-ID \(pid\) and computes \(b = H_3(M_i)\) and \(R_A = b \cdot P\), where \(M_i = \{pid, ID_{AP} \| ts\}\). Next, the MN calculates the AP’s public key \((PK_{AP} = PK_2 + H_3(ID_{AP} \| PK_2)PK_1)\), shared secret \((K_i = S_{pid} \cdot R_{AP} + c \cdot PK_{AP})\), and session key \((PTK_i = H_3(pid_i \| ID_{AP} \| K_i))\).

The AP computes the authentication messages \(H_{3}(PTK_i \| R_{AP})\) and sends \(\{M_i, R_{MN}, H_{3}(PTK_i \| R_{AP})\}\) to the AP.

2) \(AP \rightarrow MN: H_{3}(PTK_i \| R_{MN})\)

Upon receiving \(\{M_i, R_{MN}, H_{3}(PTK_i \| R_{AP})\}\), the AP initially validates the timestamp \(ts\) and terminates the session if the \(ts\) is invalid. Otherwise, the AP computes \(b = H_3(M_i)\) and \(R_{AP} = b \cdot P\). The AP then calculates MN’s public key \((PK_{MN} = PK_2 + H_3(pid_i \| PK_2)PK_1)\), shared secret \((K_i = S_{pid} \cdot R_{MN} + b \cdot PK_{MN})\), and session key \((PTK_i = H_{3}(pid_i \| ID_{AP} \| K_i))\). The AP then verifies the following equation: \(H_{3}(PTK_i \| R_{AP}) = H_{3}(PTK_i \| R_{AP})\). If the result is negative, then this message is not sent. Otherwise, the AP computes the authentication message \(H_{3}(PTK_i \| R_{MN})\) and sends it to the MN.

3) \(MN\): Verification

After receiving \(\{H_{3}(PTK_i \| R_{MN})\}\) from the AP, the MN generates \(H_{3}(PTK_i \| R_{MN})\) and compares it with the received \(H_{3}(PTK_i \| R_{MN})\). The authentication fails if they are unequal. Otherwise, the MN believes that the AP is a legal access point and a secure communication channel is erected between them.

### 3. Impersonation Attack on Xu et al.’s Protocol

In this section, we prove that the Xu et al.’s scheme cannot support the mutual authentication of communication parties and is suffer from the mobile node impersonation attack. A malicious attacker can gain access to the network service from an AP by pretending as a legal user. Furthermore, a legal MN can use a randomly selected identity to deceive an AP, which results in completely untraceability and will become unable to meet the security requirement of conditional privacy preservation [4].

The details are as follows.

**Step1:** Assume \(A\) be the adversary. When \(A\) roams into the coverage range of an AP, \(A\) randomly selects an identity \(ID_A\), generates a timestamp \(ts\), and computes the following:

\[
\begin{align*}
M_i & = \{ID_A \| ID_{AP} \| ts\}, \\
R_A & = R_{AP} = b \cdot P, \\
PK_{AP} & = PK_2 + H_3(ID_{AP} \| PK_2)PK_i, \\
PK_A & = PK_2 + H_3(ID_A \| PK_2)PK_i, \\
K_i & = b \cdot PK_{AP} + PK_{AP}, \\
PTK_i & = H_3(ID_A \| ID_{AP} \| K_i)
\end{align*}
\]

**Step2:** \(A\) sends \(\{M_i, R_A, H_{3}(PTK_i \| R_{AP})\}\) to AP.

**Step3:** AP verifies the message as follows:

\[b = H_3(M_i)\]
\[ R_{AP} = b \cdot P \]
\[ PK_A = PK_2 + H_2(ID_A \| PK_2) \]
\[ K_2 = S_{AP} \cdot R_A + b \cdot PK_A \]
\[ PTK_2 = H_2(ID_A \| ID_{AP} \| K_2) \]

Step 4: AP checks whether \( H_2(PTK_2 \| R_{AP}) \) is equal to \( H_2(PTK_1 \| R_{AP}) \). If equal, A is authenticated by the AP. Otherwise, the message is dropped.

The login request message \( \{ M_i, R_i, H_2(PTK_1 \| R_{AP}) \} \) of A can successfully pass the identity verification because:

\[ K_2 = S_{AP} \cdot R_A + b \cdot PK_A \]
\[ = S_{AP} \cdot b \cdot P + b \cdot PK_A \]
\[ = b \cdot (PK_A + PK_3) \]
\[ = K_1 \]

The AP believes A is a legal MN and provides it network service. However, A is an adversary.

Therefore, the mutual authentication and resistance of impersonation attack cannot be achieved in Xu et al.’s protocol.

4. Proposed handover authentication protocol

In this section, we design a new improvement to AMNHA protocol over Xu et al.’s approach.

The proposed protocol also contains two phases:

4.1. Pre-deployment phase

The pre-deployment phase has following two processes:

1) System initialization

The AS initially performs these four steps:

a) Selects a k-bit prime \( q \) and chooses the tuple \( \{ F_q, E/F_q, G, P \} \), where \( F_q \) is a finite field, \( E/F_q \) is an elliptic curve \( E \) over \( F_q \). \( G \) is a cyclic additive group with order \( q \), and \( P \) is a base point of \( G \) over \( E/F_q \).

b) Selects two random values \( s, r \in Z_q^* \) and uses them as system private keys. Then, calculates the corresponding public keys as \( PK_j = s \cdot P, PK_k = r \cdot P \).

c) Selects four cryptographic hash functions, \( H_j: \{0, 1\}^* \rightarrow Z_q^*, H_2: \{0, 1\}^* \rightarrow \{0, 1\}^k, H_3: \{0, 1\}^* \rightarrow \{0, 1\}^k, H_4: \{0, 1\}^* \rightarrow \{0, 1\}^* \rightarrow \{0, 1\}^k \).

d) Publishes the tuple \( \{ F_q, E/F_q, G, P, PK_j, PK_k, H_j, H_2, H_3, H_4 \} \) and keeps the system private keys secret.

2) Key pre-distribution

a) For MN, the AS selects a set of independent pseudo-IDs (PID = pid1, pid2, ...) to replace the MN’s real identity (i.e. IDMN). Then, the AS calculates the private key \( S_{pid_i} = r + H_i(pid_i \| PK_2)s \) for each pseudo-ID (pid_i \in PID) and transfers all tuples (pid_i, S_{pid_i}) to the MN via a security tunnel. The MN can generate the corresponding public key \( (PK_{pid_i} = S_{pid_i} \cdot P = PK_2 + H_i(pid_i \| PK_2)PK_1) \). To achieve user anonymity, each key pair and the corresponding pseudo-ID are only used once in the handover authentication phase.

b) For AP with real identity ID_{AP}, the AS computes its private key as \( S_{AP} = r + H_i(ID_{AP} \| PK_2)s \); then the AP’s public key \( PK_{AP} = S_{AP} \cdot P = PK_2 + H_i(ID_{AP} \| PK_2)PK_1 \). Finally, the AS sends the secret key \( S_{AP} \) to the AP through a secure channel.

4.2. Handover Authentication Phase

This phase is performed when the MN leaves the current associated AP coverage and roams into another AP. Suppose each AP broadcasts a beacon message periodically, which contains its identity and other relevant network information. The MN initially selects a target AP and extracts the identity from
the beacon message, and a handover authentication phase is then executed with AP. The steps of the handover authentication phase are described in Fig. 1.

The details are as follows.

1) \( \text{MN} \rightarrow \text{AP} \): \( \{M_i, H_3(\text{PTK}_1 \parallel R_{\text{AP}})\} \)

The MN generates a random integer \( a \in Z'_q \), a timestamp \( ts \), and computes \( R_{\text{MN}} = a \cdot P \). Then, the MN selects an unused pseudo-ID \( \text{pid}_i \) and computes \( M_i = \{\text{pid}_i \parallel ID_{\text{AP}} \parallel R_{\text{MN}} \parallel ts\} \). Subsequently, the MN calculates the AP's public key (\( PK_{\text{AP}} = PK_2 + H(ID_{\text{AP}} \parallel PK_2) \cdot PK_1 \)), shared secret (\( K_1 = (S_{\text{pid}} + a)(R_{\text{AP}} + PK_{\text{AP}}) \)), and session key (\( \text{PTK}_1 = H_2(\text{pid}_i \parallel ID_{\text{AP}} \parallel K_1) \)).

Finally, the MN computes the authentication messages (\( H_3(\text{PTK}_1 \parallel R_{\text{AP}}) \)) and sends the login request message \( \{M_i, H_3(\text{PTK}_1 \parallel R_{\text{AP}})\} \) to AP.

2) \( \text{AP} \rightarrow \text{MN} \): \( H_3(\text{PTK}_2 \parallel R_{\text{MN}}) \)

Upon receiving \( \{M_i, H_3(\text{PTK}_1 \parallel R_{\text{AP}})\} \), the AP initially checks whether \( ts \) is fresh and computes \( b = H_4(M_i) \) and \( R_{\text{AP}} = b \cdot P \) or otherwise terminates the session. Next, the AP generates the MN's public key (\( PK_{\text{MN}} = PK_2 + H_3(\text{pid}_i \parallel ID_{\text{AP}} \parallel PK_2) \cdot PK_1 \)), shared secret (\( K_2 = (S_{\text{AP}} + b)(PK_{\text{MN}} + R_{\text{MN}}) \)), and session key (\( \text{PTK}_2 = H_2(\text{pid}_i \parallel ID_{\text{AP}} \parallel K_2) \)).

The AP then checks whether \( H_3(\text{PTK}_2 \parallel R_{\text{MN}}) \) is the same as the received \( H_3(\text{PTK}_1 \parallel R_{\text{AP}}) \). If the two values are not the same, then the authentication will fail. Otherwise, the AP computes the authentication message \( H_3(\text{PTK}_2 \parallel R_{\text{MN}}) \) and sends it to the MN as a response message.

3) \( \text{MN} \rightarrow \text{AP} \):

Upon receiving the message \( H_3(\text{PTK}_2 \parallel R_{\text{MN}}) \), the MN initially computes \( H_3(\text{PTK}_1 \parallel R_{\text{MN}}) \) and compares it with \( H_3(\text{PTK}_2 \parallel R_{\text{MN}}) \). If the result is negative, then the session is terminated. Otherwise, the MN believes that AP is a legal service provider.

Once the mutual authentication is finished, a secure communication channel is established between two communicating parties using the shared session key \( \text{PTK}_1 \) or \( \text{PTK}_2 \).

5. Security Analysis

5.1. Mutual Authentication and key Agreement

In our protocol, the two communicating parties (i.e., MN and AP) can authenticate each other because they share the same secret:
\[ K_i = (S_{pid} + a)(PK_{AP} + R_{AP}) \]
\[ = (S_{pid} + a)(R_{AP} + H_2(pid)\parallel PK_2PK_1 + R_{AP}) \]
\[ = (S_{pid} + a) \cdot P \cdot (S_{AP} + b) \]
\[ = (PK_{MN} + R_{MN})(S_{AP} + b) \]
\[ = K_2 \]

Hence, MN and AP can calculate the same session key
\[ PTK_1 = H_2(pid_1 \parallel ID_{AP} \parallel K_i) \]
\[ = H_2(pid_1 \parallel ID_{AP} \parallel K_2) \]
\[ = PTK_2 \]

Based on the Computational Diffie-Hellman (CDH) problem[17] and Decisional Diffie-Hellman (DDH) problem[18], no one can forge the private key of MN and AP and generate the shared \( K_i \) or \( K_2 \) secret, as well as the session key. Only the legitimate MN and AP can pass the identity authentication through the valid \( H_2(PTK_1 \parallel R_{AP}) \) and \( H_2(PTK_2 \parallel R_{MN}) \) and establish the shared session key for both parties.

Consequently, the proposed protocol supports mutual authentication for both parties of communication, as well as the key establishment between MN and AP.

5.2. User Anonymity and Untraceability
To protect privacy, the AS generates a family of unlinked pseudo-IDs for each MN, which is used to replace the real identity during the handover authentication stage. Hence, only the AS knows the correspondence between the real identity of the MN and the pseudo-IDs. Furthermore, given that the login request message \( \{M_i, H_3(PTK_1 \parallel R_{AP})\} \) will constantly change because the number \( a \) is randomly selected by the MN during each session run, an attacker can obtain the pseudo-ID through the \( M_i \). However, the attacker cannot extract the real identity of the MN from the pseudo-ID and trace the behavior of the MN given that no linkage between them is present.

Hence, our scheme can provide user anonymity and untraceability.

5.3. Perfect Forward Secrecy
In the proposed protocol, a temporal Diffie–Hellman parameter (i.e., \( a \)), which is selected randomly by the MN during each session run, is used to generate the shared secret and session keys and further establish the security communication channel between the MNs and APs. Hence, no relation exists between the session key and other session keys or the long-term system private key. Furthermore, even though the latter are compromised by an attacker, the preceding session keys cannot be obtained and the transmitted information between the MNs and APs cannot be extracted.

Therefore, the new protocol can provide perfect forward secrecy.

5.4. Resist Impersonation Attacks
The private keys of the MN and the AP are computed by the AS based on the pseudo-IDs of the MN and the identity of the AP, respectively, and then delivered through a secure channel. Hence, without the system private key, an attacker cannot generate a valid public/private key pairs, and launch impersonation attacks by forging the legitimate authentication message.

In addition, any changes in \( M_i \) will result in a series of values change, such as \( b, R_{AP}, K_2, PTK_2 \) and \( H_3(PTK_2 \parallel R_{AP}) \). Thus, the AP can easily detect the impersonation attacks by checking \( H_3(PTK_2 \parallel R_{AP}) \).

Without the private key of the AP, an attacker cannot generate a valid \( K_2 \) and \( PTK_2 \); thus, the MN can easily find this type of attack by comparing the received \( H_3(PTK_2 \parallel R_{MN}) \).

Thus, our protocol can effectively prevent the impersonation attack.

5.5. Resist Replay Attacks
Our protocol can efficiently withstand message replaying attacks given that each login request message includes a time stamp \( ts \). Thus, the AP can easily detect this type of attack by checking whether the \( ts \) is fresh.
Furthermore, an adversary, including malicious APs, cannot replay this message to other APs even the time ts is fresh due to the login request message that contains the identities of two communication parties.

6. Performance evaluation

In this section, we briefly compare the security properties and performance of our protocol with three recent protocols.

6.1 Comparison of security properties

Table 1 provides a summary and comparison of the security properties of our improved AMNHA protocol and other relevant works [2,14,15].

The data presented in Table 1 indicate that He et al.’s scheme[2] cannot prevent key compromise attacks, whereas Xu et al.’s scheme[15] cannot resist impersonation attacks and provide mutual authentication. The proposed AMNHA protocol and Xie et al.’s protocol satisfies all the security requirements, thereby making them more secure and efficient than other two schemes and suitable for wireless networks.

6.2 Comparison of computation costs

To illustrate the performance comparisons, some of the symbols are defined below:
- $T_{BP}$: Time cost for a bilinear pairing operation
- $T_{PM}$: Time cost for a scalar point multiplication based on pairing
- $T_{PA}$: Time cost for a point addition based on pairing
- $T_{MTP}$: Time cost for a map-to-point hash operation
- $T_{EPM}$: Time cost for an elliptic curve scalar point multiplication
- $T_{EPA}$: Time cost for an elliptic curve point addition

Table 1 Comparison of security properties

|                          | He[2] | Xu[15] | Xie[14] | Ours |
|--------------------------|-------|--------|---------|------|
| Mutual authentication    | YES   | NO     | YES     | YES  |
| User anonymity and untraceability | YES   | YES    | YES     | YES  |
| Perfect forward secrecy  | YES   | YES    | YES     | YES  |
| Resist impersonation attack | YES   | YES    | YES     | YES  |
| Resist replay attack     | YES   | NO     | YES     | YES  |
| Resist key compromised attack | NO    | YES    | YES     | YES  |

Table 2 Running time of related operations (ms)

|       | $T_{BP}$ | $T_{PM}$ | $T_{PA}$ | $T_{MTP}$ | $T_{EPM}$ | $T_{EPA}$ |
|-------|----------|----------|----------|-----------|-----------|-----------|
| AP    | 1.212    | 0.001    | 0.005    | 0.135     | 0.458     | 0.004     |
| MN    | 3.729    | 0.002    | 0.015    | 0.375     | 1.451     | 0.019     |

Table 3 Comparisons of computation cost (ms)

|       | AP         | MN         | Total Time |
|-------|------------|------------|------------|
| He[2] | $3T_{BP} + T_{PM} = 3.637$ | $T_{BP} + T_{PM} = 3.731$ | 7.368 |
| Xu[15] | $4T_{EPM} + 2T_{EPA} = 1.84$ | $5T_{EPM} + 2T_{EPA} = 7.293$ | 9.133 |
| Xie[14] | $6T_{EPM} + 3T_{EPA} = 2.76$ | $5T_{EPM} + 2T_{EPA} = 7.293$ | 10.053 |
| Ours  | $3T_{EPM} + 2T_{EPA} = 1.382$ | $4T_{EPM} + 2T_{EPA} = 5.842$ | 7.224 |

- $T_{PM}$: Time cost for a scalar point multiplication based on pairing
- $T_{PA}$: Time cost for a point addition based on pairing
- $T_{MTP}$: Time cost for a map-to-point hash operation
- $T_{EPM}$: Time cost for an elliptic curve scalar point multiplication
- $T_{EPA}$: Time cost for an elliptic curve point addition

Table 2 shows the execution time of the bilinear pairing and elliptic curve operations. We assume that an AP runs on an Intel Core i5 2.4 GHz processor and an MN runs on 1 GHz processor, and the operating system is Ubuntu 16.04 with 2 GB of RAM. The running time of the bilinear pairing operations and elliptic curve operations is based on the PBC[19] library and MIRACL[20] library. We
use the MNT family of curves[21] with an order of 160 bits and the embedding degree $k = 6$ for ECC operation.

A comparison of the computation costs between the new protocol and other three relevant protocols are listed in Table 3. Table 3 shows that the total computation cost of our protocol is 7.224 ms, and, which is a slightly less than the total time of He et al.’s protocol[2].

In comparison with the schemes proposed by Xu et al. and Xie et al.’s, the total execution time of MN and AP in our protocol has decreased by 20.9% and 28.14%, respectively.

6.3 Comparison of communication cost
In the simulation, we assume that the security level of ECC and the bilinear pairing is 80 bits. Thus, the size of $p$ is 64 bytes, and the element in $G$ (additive group in bilinear pairing) is 128 bytes; the size of $q$ is 20 bytes, so the element in $G$ (ECC) is 40 bytes. We suppose that the length of the output of all hash functions and all identities are 20 bytes, and the length of the timestamp is 4 bytes. The transmitted messages between MN and AP in our protocol are $\{M_i, H(PTK_i) || R_{ID}\}$ and $\{H(PTK_i) || R_{MN} || ts\}$; thus, the total size is 124 bytes. We can similarly compute the communication cost of the scheme of He et al., Xu et al., and Xie et al. as 232 bytes, 124 bytes, and 268 bytes, respectively.

Our proposed protocol and Xu et al.’s have the same value, which are reduced by 8% and 53.7% when compared with the protocol of He et al. and Xie et al., respectively.

7. Conclusion
In this paper, we reviewed the recent studies on MWNs, and found that these AMNHA protocols can be divided into two types; that is, one that is based on bilinear pairing operations and ECC operations, and the other is only based on ECC operations. We then analyzed the pairing-free protocol of Li et al. and found that their scheme suffers from impersonation attacks. Through a randomly selected identity, an adversary (or legitimate MNs) can access the network service by cheating the AP. To address the above problem, we present an improved AMNHA protocol for MWNs. The proposed protocol not only satisfies the entire security properties and is robust against network threats but also has a lower computation and communication cost when compared to other recent protocols.

8. Acknowledgment
This work was supported in part by the National Natural Science Foundation of China under grants U1435213 and 61172180, the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 2012018110053, and the Youth Science and Technology Foundation of Sichuan Province under Grant 2013JQ0002.

9. References
[1] D. He, C. Chen, S. Chan and J. Bu. “Strong roaming authentication technique for wireless and mobile networks,” International Journal of Communication Systems, vol. 26, no. 8, 2013, pp. 1028-1037.
[2] D. He, C. Chen, S. Chan and J. Bu, “Secure and Efficient Handover Authentication Based on Bilinear Pairing Functions,” IEEE Transactions on Wireless Communications, vol. 11, no. 1, Jan. 2012, pp. 48-53.
[3] D. He, C. Chen, S. Chan and J. Bu, “Analysis and Improvement of a Secure and Efficient Handover Authentication for Wireless Networks,” IEEE Communications Letters, vol. 16, no. 8, 2012, pp. 1270-1273.
[4] D. He, S. Chan and M. Guizani. “Handover authentication for mobile networks: security and efficiency aspects,” IEEE Network, vol. 29, no. 3, 2015, pp. 96-103.
[5] D. He, D. Wang, Q. Xie and K. Chen. “Anonymous handover authentication protocol for mobile wireless networks with conditional privacy preservation,” Science China Information Sciences, vol. 60, no. 5, 2017, pp. 052104.
[6] D. He, M. K. Khan and N. Kumar. “A new handover authentication protocol based on bilinear pairing functions for wireless networks,” International Journal of Ad Hoc & Ubiquitous Computing, vol. 18, no.1/2, 2015, pp. 67-74.

[7] W. Wang, L. Hu. “A secure and efficient handover authentication protocol for wireless networks,” Sensors, vol. 14, no. 7, 2014, pp. 11379-11394.

[8] J. L. Tsai, N. W. Lo and T. C. Wu. “Secure handover authentication protocol based on bilinear pairings,” Wireless Personal Communications, vol. 73, no. 3, 2013, pp. 1037-1047.

[9] X. Cao, W. Kou and X. Du. “A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges”. Information Sciences, vol. 180, n. 15, 2010, pp. 2895-2903.

[10] S. H. Islam, G. P. Biswas. “An improved pairing-free identity-based authenticated key agreement protocol based on ECC”. Procedia Engineering, vol. 30, no.4, 2012, pp. 499-507.

[11] J. Cao, M. Ma and H. Li. “An Uniform Handover Authentication between E-UTRAN and Non-3GPP Access Networks,” IEEE Transactions on Wireless Communications, vol. 11, no. 10, 2012, pp. 3644-3650.

[12] G. Li, Q Jiang and F Wei and C. Ma “A New Privacy-Aware Handover Authentication Scheme for Wireless Networks,” Wireless Personal Communications, vol.80, no. 2, 2015, pp. 581-589.

[13] S. A. Chaudhry, M. S. Farash, H. Naqvi, S. H. Islam and T. Shon. “A Robust and Efficient Privacy Aware Handover Authentication Scheme for Wireless Networks”. Wireless Personal Communications, vol.93, no. 2, 2017, pp. 1-25.

[14] Y. Xie, L. Wu, N. Kumar and J. Shen. “Analysis and Improvement of a Privacy-Aware Handover Authentication Scheme for Wireless Network”. Wireless Personal Communications, vol. 93, no.2, 2017, pp. 523-541.

[15] Y. Xu, Y. Zhang, J. K. Liu and Y. Zeng. “A Trust and Privacy Preserving Handover Authentication Protocol for Wireless Networks,” Trustcom/BigDataSE/ISPA, 2016, pp. 138-143, DOI: 10.1109/TrustCom.2016.0056

[16] X. Cao, X. Zeng, W. Kou, L. Hu, “Identity-based anonymous remote authentication for value-added services in mobile networks,” IEEE Transactions on Vehicular Technology, vol. 58, no. 7, 2009, pp. 3508-3517.

[17] I. Shparlinski. “Computational Diffie-Hellman Problem,” Encyclopedia of Cryptography and Security, 2011, pp. 240-244, DOI 10.1007/978-1-4419-5906-5_882.

[18] C. Ran. “Decisional Diffie-Hellman Assumption,” Encyclopedia of Cryptography & Security, 2005, pp. 140-142, DOI 10.1007/0-387-23483-7_99.

[19] Pairing based cryptography benchmarks. (Available from: http://crypto.stanford.edu/pbc/times.html).

[20] S. Michael. “Multiprecision Integer and Rational Arithmetic C/C++ Library(MIRACL),” Published by Shamus Software Ltd., http://www.shamus.ie/.

[21] A. Miyaji, M. Nakabayashi and S. Takano. “New explicit conditions of elliptic curve trace for FR-reduction,” IEICE Transactions on Fundamentals, vol. 84, no. 5, 2001, pp. 1234-1243.