International Journal of Computer Science and Business Informatics

IJCSBI.ORG

# An Efficient Authentication Protocol for Mobile Cloud Environments using ECC

**Mohammad Rasoul Momeni**

Department of Computer Engineering, Faculty of Electrical and Computer Engineering,
Imam Reza International University of Mashhad, Iran

## ABSTRACT

The growth of mobile cloud computing users is rapid and now many mobile users utilize from mobile cloud computing technology. This technology makes mobile users stronger beyond the mobile computing capabilities. The security risks have become a hurdle in the rapid adaptability of the mobile cloud computing technology. Significant efforts have been devoted in research organizations and academia to securing the mobile cloud computing technology. In this paper we proposed a lightweight and efficient authentication protocol for mobile cloud environment. According to significant advantages of ECC (elliptic curve cryptosystem), it has been adopted through this paper. Our proposed protocol has many advantages such as: supporting user anonymity, identity management and also resistance against related attacks such as replay attack, stolen verifier attack, modification attack, server spoofing attack and so on.

### Key words

Mobile cloud computing, lightweight authentication, ECC, user anonymity, security risks.

## 1. INTRODUCTION

Mobile cloud computing is a technology that aims to augment mobile devices beyond their capabilities. Mobile devices have limited processing and storage capabilities and their battery lifetime will exhaust soon [1]. Authentication is the most important factor to protect systems against attacks. If this mechanism works well other mechanisms can be lightweight. Authentication methods are grouped to four classes. 1. What you are (E.g. fingerprint), 2.what you have (E.g. smart cards), 3.what you know (E.g. passwords) and 4.what you do or implicit authentication. Due to inherent challenges of wireless communications such as insecure nature and problems related to heterogeneity, security and privacy issues are too complex in mobile cloud computing. ECC based schemes with smaller key size, strict security and high efficiency are the best choice for securing the mobile cloud computing technology. For example ECC with 160 bits key size and RSA with 1024 bits key size have the same security level. ECC is good for environments with these properties: low bandwidth, limited processing power and storage capacity, battery lifetime limitation. Firstly



lamport in 1981 proposed an authentication scheme over an open channel [2]. His scheme was resistance against impersonation attack and server's data eavesdropping attack but vulnerable to replay attack. Peyravian and Zunic proposed an authentication scheme without encryption techniques [3]. It only used hash function. Lee et al. demonstrated that this scheme is vulnerable to offline password guessing attack and then improved it [4]. Later ku et al showed lee et al scheme is vulnerable to attacks such as denial of service, offline password guessing and stolen verifier [5]. Yoon et al then improved lee et al scheme in the year 2004 [6], but ku et al demonstrated this scheme is vulnerable to offline password guessing attack and stolen verifier attack [7]. Later hwang and yeh demonstrated Peyravian and Zunic's scheme is vulnerable to password guessing attack and server spoofing attack [8]. Then they improved it with public key cryptosystem. Their scheme satisfies mutual authentication, but ku et al mentioned it is vulnerable to replay attack [9]. Lin and hwang demonstrated denial of service attack is applicable to hwang and yeh scheme [10]. Also they mentioned hwang and yeh scheme cannot satisfy perfect forward secrecy. Peyravian and Jeffries improved Peyravian and Zunic's scheme [11], but shim claimed that their scheme is vulnerable to offline password guessing attack and denial of service attack [12]. Zhu et al mentioned that Hwang and Yeh's scheme still vulnerable to replay attack, stolen verifier attack and impersonation attack and then proposed an improved scheme to eliminate the weaknesses of Hwang and Yeh's scheme [13]. Their scheme is based on timestamp and salting techniques. Momeni proposed a lightweight authentication scheme [14]. His protocol has little processing and communication overhead and is enough strong against related attacks. The rest of the paper is organized as follows: in Section 2, we propose our authentication protocol. Section 3 and 4 describe the security and performance analysis respectively. And finally section 5 concludes the paper. The notations to be used in this paper are in Table 1.



**Table 1. Notations**

| Symbols | Description |
|---|---|
| $ID_U$ | User identity |
| $PW_U$ | Password |
| $S$ | Private key (server) |
| $Q = S.P$ | Public key (server) |
| $AK_U = S.Z_U$ | Authentication key |
| $Z_U = PW_U.P$ | Password verifier |
| $DID_U$ | Dynamic identity of user |
| $P$ | Base point |
| $\|\|$ | Concatenation operator |
| $H(\ )$ | Hash function |
| $r_1$ , $r_2$ | Random numbers |
| $E_{AK}(\ )$ | Symmetric encryption function |

## 2. PROPOSED SCHEME

In this section our protocol is presented. Our proposed protocol consists of four phases namely: registration phase, mutual authentication and session key agreement phase, password change phase and finally user eviction phase. Now we describe the registration phase.

### 2.1 Registration Phase

In this phase mobile user performs registration phase via secure channel as follows. Note that registration phase is done only once.

1. Mobile user sends his identity and password verifier to authentication server through secure channel.

2. Now the server checks this identity and if already exists in the server database rejects it, Mobile user must prepares unique identity. It is clear to see that in this step identity management is provided. Now server can compute authentication key as $AK_U = S.Z_U$. In this step authentication server stores user identity, password verifier and status bit in the users table.

3. Finally the server returns authentication key to mobile user.

When mobile user login to system, status bit sets to one and in the other words it sets to zero. In the following we show a sample of users table.



**Table 2. Users table**

| Identity | Password Verifier | Status bit |
|---|---|---|
| $ID_A$ | $Z_A = PW_A.P$ | 0-1 |
| $ID_B$ | $Z_B = PW_B.P$ | 0-1 |
| $ID_C$ | $Z_C = PW_C.P$ | 0-1 |

Registration phase has shown in figure 1.

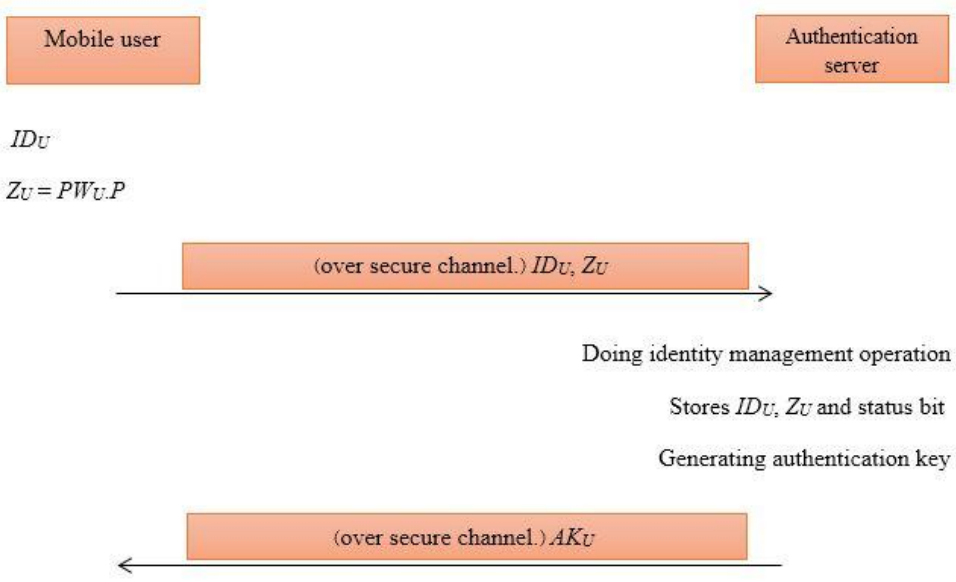



**Figure 1. Registration phase**

## 2.2 Mutual authentication and session key agreement phase

After registration whenever mobile user wants to use cloud services, he/she must be authenticated. Hence he/she sends a login request message to the server and then server verifies the authenticity of the login request message as follows.

1. Mobile user generates a random number $r_1$ and calculates $R = r_1.Q$ and also $M = r_1.PW_U.P$. Next mobile user generates dynamic identity to protect his real identity as follows: $DID_U = ID_U \oplus H(AK_U \| R)$. Mobile user sends $M_1 = (DID_U, E_{AK}(R, M), H(DID_U, E_{AK}(R, M)))$ to the server.

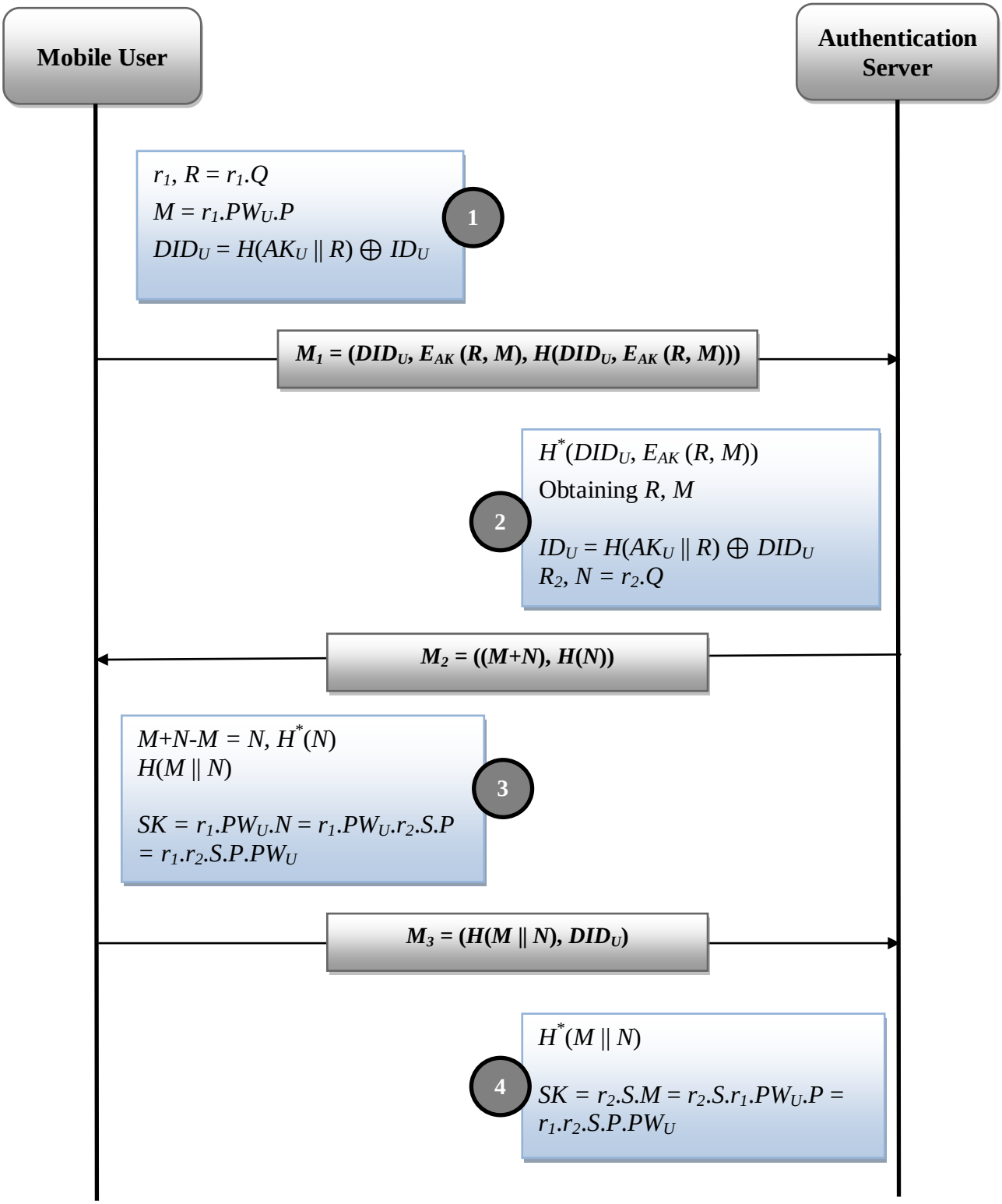
2. After receiving $M_1$, the server computes $H^*(DID_U, E_{AK}(R, M))$, then checks $H = H^*$ for detecting modification attack. If $H$ is not equal to $H^*$, server aborts the current session. Hence denial of service can be eliminated.



Then server obtains $R$ and $M$ by decrypting the message. Also in this step server calculates the real identity from dynamic identity as follows: $ID_U = DID_U \oplus H(AK_U \| R)$. Then validates it according to identities that exists in the users table. Now the server generates $r_2$ and calculates $N = r_2.Q$. Finally the server sends message $M_2 = ((M+N), H(N))$ to mobile user.

3. After receiving $M_2$, mobile user calculates $N$ from $M+N-M$ and then computes $H^*(N)$ and compares it by received $H(N)$ to detect modification attack. If $H$ is not equal to $H^*$ mobile user aborts the current session, Hence denial of service can be eliminated. Mobile user computes message $M_3 = (H(M \| N), DID_U)$ and sends it to the server. Also he/she computes the session key $SK = r_1.PW_U.N = r_1.PW_U.r_2.S.P = r_1.r_2.S.P.PW_U$ in this step.

4. After receiving $M_3$, the server computes $H^*(M \| N)$ and then compares it by received $H(M \| N)$ to detect modification attack. If $H$ is not equal to $H^*$ the server aborts the current session, Hence denial of service can be eliminated. Now the server computes session key as follows: $SK = r_2.S.M = r_2.S.r_1.PW_U.P = r_1.r_2.S.P.PW_U$. Note that $SK$ is valid only for this session. Mutual authentication and session key agreement phase has shown in following.


Mobile User
Authentication Server
$r_1$, $R = r_1.Q$
$M = r_1.PW_U.P$
$DID_U = H(AK_U \| R) \oplus ID_U$
1
$M_1 = (DID_U, E_{AK}(R, M), H(DID_U, E_{AK}(R, M)))$
$H^*(DID_U, E_{AK}(R, M))$
Obtaining $R$, $M$
$ID_U = H(AK_U \| R) \oplus DID_U$
$R_2$, $N = r_2.Q$
2
$M_2 = ((M+N), H(N))$
$M+N-M = N$, $H^*(N)$
$H(M \| N)$
$SK = r_1.PW_U.N = r_1.PW_U.r_2.S.P = r_1.r_2.S.P.PW_U$
3
$M_3 = (H(M \| N), DID_U)$
$H^*(M \| N)$
$SK = r_2.S.M = r_2.S.r_1.PW_U.P = r_1.r_2.S.P.PW_U$
4

### 2.3 Password change phase

In this phase mobile user can change his/her password without any intervention from server. This property brings high security and user friendly for our proposed scheme. After choosing new password and computing password verifier for it, only password verifier transmits to the server. The channel in this phase is secure. The steps of this phase are as follows.

1. Mobile user sends his/her identity and password verifier with a password change request to the server.

2. After verifying the received identity if he/she is a legitimate mobile user, the server computes $H(ID_U \| SK)$ and sends it to mobile user.

3. Then mobile user computes $H^*(ID_U \| SK)$ and compares it with received $H(ID_U \| SK)$. If $H^*$ is equal by $H$ then mobile user selects new password and computes password verifier for it as follows: $Z_U^* = PW_U^*.P$. Finally only password verifier will send to the server.

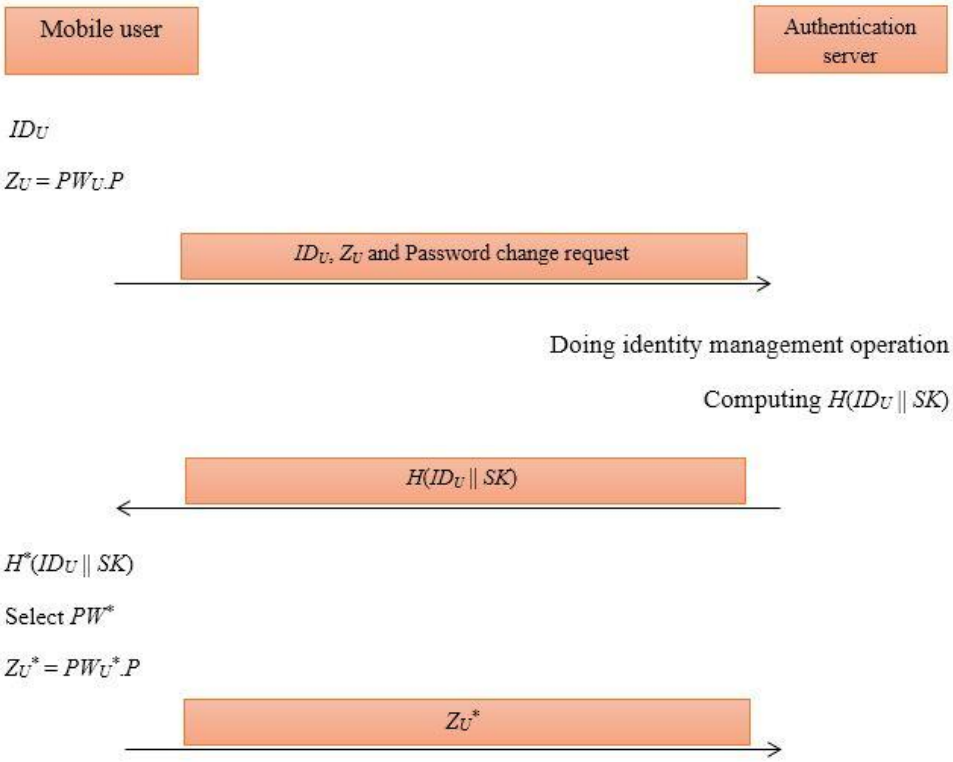



**Figure 2. Password change request**

### 2.4 User eviction phase

In the proposed protocol server can evict malicious users. In order to evict malicious users, the server should remove related tuples from users table. If an evicted user tries to login to the system, he/she will fail because in the second phase of the mutual authentication and session key agreement identity management will be conducted. Thus the server will know this identity does not exist in the users table. As a result evicted users cannot login to the system and use cloud resources.

## 3. SECURITY ANALYSIS

In this section security features of our proposed protocol is presented and we demonstrate proposed protocol can withstand against related security attacks.

### 3.1 Stolen verifier attack resistance

Our proposed protocol is robust against stolen verifier attack because server does not keep any secret table or any pre-shared secret key. Hence adversary cannot gain any valuable information from this attack.

### 3.2 Server spoofing attack resistance

Our proposed protocol provides mutual authentication for both participants. Mobile user authenticates the server and also server can authenticate the mobile user. Hence sever spoofing attack is ineffective.

### 3.3 Modification attack resistance

In order to avoid modification attack we used a collision free one way hash function. If an adversary sends a modified message, soon mobile user will know that the received message is not valid because two hash results are not equal.

### 3.4 Replay attack resistance

Proposed scheme uses random numbers to avoid replay attack. It is hard for adversary to guess the random numbers, because they change in each session and every time of authentication. Thus this attack is not applicable to our scheme.

### 3.5 Insider attack resistance

A client CL may register with some servers $S_1$, $S_2$ and so on using a common password pw and the identity id for his convenience, and if the privileged-insider $U_1$ of $S_1$ has the knowledge of CL's pw and id, then $U_1$ may try to access other servers $S_2$, $S_3$, and so on by using the same pw and id. In our proposed protocol the server only stores password verifier and



extraction of password from it is very hard due to hardness of elliptic curve discrete logarithm problem (ECDLP).

### 3.6 Many logged-in users attack resistance

Consider the password $PW_A$ and the login-id $ID_A$ of a client A, are leaked to many adversaries. In the proposed protocol only one adversary can login the server at the same time out of all who know the valid password $PW_A$ and login-id $ID_A$. When an adversary logged-in by using the valid password $PW_A$ and login-id $ID_A$, then the server sets the status bit to one and meanwhile if other adversaries try to login the server at the same time with same password $PW_A$ and login-id $ID_A$, the server denies all the received requests because the status bit mechanism indicates still someone is logged in.

### 3.7 User anonymity

User anonymity means protecting real identity of user against public, no server [15]. Our proposed scheme satisfies user anonymity, because in the registration and password change phases that real identity transmits, the channel is secure and in the mutual authentication and session key agreement phase that channel is not secure instead of real identity, dynamic identity transmits.

### 3.8 No clock synchronization problem

Many proposed authentication protocols use timestamps to avoid replay attacks but timestamp mechanism is difficult and expensive in wireless mobile communications [16] and distributed networks [17,18,19]. Our proposed protocol is nonce-based and does not have clock synchronization problem.

### 3.9 Session key agreement

In our proposed protocol a session key is generated which uses random numbers like $r_1$ and $r_2$. This session key provides secure communications over open channel by encrypting the exchanged messages.

### 3.10 Password change phase

Our proposed protocol supports Password change phase, hence our protocol is more secure than other authentication protocols. In addition mobile user can change his/her password without any intervention from server. This property brings high security and user friendly for our proposed scheme.

## 4. PERFORMANCE ANALYSIS

In this section we evaluate the performance of our proposed protocol. Note that a good authentication scheme for mobile cloud computing must have low computation cost. We compared our proposed protocol with a new proposed scheme in terms of computation cost. Table 3 shows that our proposed protocol is more efficient, especially in the mobile user side. This improvement makes less power consumption in the mobile devices which are faced with battery lifetime limitation.

| Hafizful & Biswas [20] | **Proposed Protocol** |
|---|---|
| Client: 7EM + 2EA + 1sym<br>Server: 3EM + 2EA + 1sym | Client: 5EM + 1EA + 1sym<br>Server: 3EM + 1EA + 1sym |

## 5. CONCLUSIONS

In this paper we proposed a lightweight authentication protocol for mobile cloud computing. In the proposed protocol we used elliptic curve cryptosystem which has many advantages includes smaller key size, strict security and high efficiency. Also our proposed protocol satisfies user anonymity, mutual authentication, session key agreement and so on. In terms of resistance against related attacks, our proposed protocol is robust against replay attack, stolen verifier attack, modification attack, server spoofing attack and so on. It is important to note that proposed protocol is according to real communication scenarios.

## REFERENCES

[1]. Momeni, M. R., 2015. A Survey of Mobile Cloud Computing: Advantages, Challenges and Approaches. International Journal of Computer Science and Business Informatics, special issue: Vol. 15, No. 4, pp. 14-28.

[2]. L. Lamport, Password authentication with insecure communication, Communications of the ACM 24 (11) (1981) 770–772.

[3]. M. Peyravian, N. Zunic, Methods for protecting password transmission, Computers and Security 19 (5) (2000) 466–469.

[4]. C.C. Lee, L.H. Li, M.S. Hwang, A remote user authentication scheme using hash functions, ACM Operating Systems Review 36 (4) (2002) 23–29.

[5]. W.C. Ku, C.M. Chen, H.L. Lee, Weaknesses of Lee–Li–Hwang's Hash-based password authentication scheme, ACM Operating Systems Review 37 (4) (2003) 19–25.

[6]. E.J. Yoon, E.K. Ruy, K.Y. Roo, A secure user authentication scheme using hash functions, ACM Operating Systems Review 38 (2) (2004) 62–68.

[7]. W.C. Ku, M.H. Chaing, S.T. Chang, Weaknesses of Yoon–Ryu–Yoo's hash-based password authentication scheme, ACM Operating Systems Review 39 (1) (2005) 85–89.

[8]. J.J. Hwang, T.C. Yeh, Improvement on Peyravian–Zunic's password authentication schemes, IEICE Transactions on Communications E85-B (4) (2002) 823–825.



[9]. W.C. Ku, C.M. Chen, L. Hui, Cryptanalysis of a variant of Peyravian–Zunic's password authentication scheme, IEICE Transactions on Communications E86-B (5) (2002) 1682–1684.

[10]. C.L. Lin, T. Hwang, A password authentication scheme with secure password updating, Computers and Security 22 (1) (2003) 68–72.

[11]. M. Peyravian, C. Jeffries, Secure remote user access over insecure networks, Computer Communications 29 (5) (2006) 660–667.

[12]. K.A. Shim, Security flaws of remote user access over insecure networks, Computer communications 30 (1) (2006) 117–121.

[13]. L. Zhu, S. Yu, X. Zhang, Improvement upon mutual password authentication scheme, International seminar on business and information management, 2008, pp. 400–403.

[14]. Momeni, M. R., 2014. A Lightweight Authentication Scheme for Mobile Cloud Computing. International Journal of Computer Science and Business Informatics, Vol. 14, No. 2, pp. 153-160.

[15]. D. Wanga, Chun-guang, Cryptanalysis of a remote user authentication scheme for mobile client–server environment based on ECC, Information Fusion 14 (2013) 498–503.

[16]. Giridhar, P. Kumar, Distributed clock synchronization over wireless networks: algorithms and analysis, in: Proceedings of the 45th IEEE Conference on Decision and Control, IEEE, 2006, pp. 4915–4920.

[17]. D. Mills, Internet time synchronization: the network time protocol, IEEE Transactions on Communications 39 (10) (1991) 1393–1482.

[18]. J. Han, D. Jeong, A practical implementation of IEEE 1588–2008 transparent clock for distributed measurement and control systems, IEEE Transactions on Instrumentation and Measurement 59 (2) (2010) 433–439.

[19]. R. Baldoni, A. Corsaro, L. Querzoni, S. Scipioni, S. Piergiovanni, Coupling-based internal clock synchronization for large-scale dynamic distributed systems, IEEE Transactions on Parallel and Distributed Systems 21 (5) (2010) 607–619.

[20]. SK Hafizul Islam, G.P. Biswas, Design of improved password authentication and update scheme based on elliptic curve cryptography, Mathematical and Computer Modelling 57 (2013) 2703–2717.

This paper may be cited as:

Momeni, M. R., 2015. An Efficient Authentication Protocol for Mobile Cloud Environments using ECC. *International Journal of Computer Science and Business Informatics, Special Issue: Vol. 15, No. 4, pp. 29-39.*