*Journal of Information Hiding and Multimedia Signal Processing* ©2014 ISSN 2073-4212
**Ubiquitous International** Volume **5**, Number **4**, October **2014**

# A DoS-Resistant Robust Identification and Key Agreement Protocol with User Anonymity

Kai Chain[a], Wen-Chung Kuo[b], Jyun-Pei Siang[c] and Jiin-Chiou Cheng[c]

[a]Department of Computer and Information Science,
R.O.C. Military Academy, Taiwan.

[b]Department of Computer Science and Information Engineering
National Yunlin University of Science & Technology

[c]Department of Computer Science and Information Engineering,
Southern Taiwan University, Taiwan, R.O.C.

{simonkuo@yuntech.edu.tw}

Received January, 2014; revised March, 2014
(Communicated by Wen-Chung Kuo)

Abstract. *In 2004, Yang* et al.*[11]proposed a user identification and key distribution protocol, which can derive a common secret key with among participants while providing user logging with anonymity. However, Mangipudi and Katti[6] pointed out that Yang* et al.*'s protocol[11] will suffer from denial of service attack. So they proposed new protocol to overcome the drawback. Recently, Wang* et al. *[7] also proposed an identification and key agreement protocol with user anonymity to improve upon the computation cost and communication overhead for the above protocol. Wang* et al.*'s protocol[7] may be suitable for mobile devices with restricted resources. Under limited resources, storage space and computation cost will be limiting factors for rapid communication with more servers. In this paper, we propose a modified protocol to improve Wang* et al.*'s protocol with the addition of DoS resistance and decreased communication cost.*
**Keywords:** Key agreement, Elliptic curve cryptography, Anonymity, DoS-resistant.

1. **Introduction.** When a user wants to securely access a resource or request for service, they will take the initiative to establish a login session with the server, and after mutual authentication agree to a common session key to protect subsequent exchanged messages[1,2,3,5]. In 2000, Lee and Chang [4] proposed a user identification and key distribution protocol that attains user anonymity based on public key cryptography (RSA) and hash functions over distributed environments. However, Wu and Hsu[8] pointed out that Lee *et al.*'s protocol[4] is vulnerable to server spoofing and the identity of a valid user could be exposed. So they proposed an improved method for enhancing security and efficiency. Later, Yang *et al.*[11] showed a new weakness in Wu and Hsu's protocol where a service provider could obtain a valid user's secret token after an exchange of messages. As such, Yang *et al.* proposed a protocol that overcomes the weakness of Wu and Hsu's protocol and achieves user anonymity, user identification and key agreement. In 2006, Mangipudi and Katti[6] pointed out that Yang *et al.*'s protocol possessed a vulnerability which can be exploited to launch a Denial-of Service (DoS) attack. At the same time, Mangipudi and Katti proposed a secure identification and key agreement protocol with user anonymity (SIKA)[6]. Unfortunately, Wang *et al.*[7] pointed out the aforementioned protocols[4,6,8,11] are not suitable for pervasive computing environments. Because the

user requires access to services from multiple servers, computation cost, and storage space requirements may exceed device capabilities. Apart from user anonymity, three attractive features are included in Wang*et al.*'s protocol: (1) each user only needs to maintain one secret, even for accessing several service providers; (2) the server is not required to maintain a list of passwords; (3) if a new service provider joins the system, and the user's master key does not need to be updated. In previous methods[12,13,14,15], the user and server must expend considerable computation cost, and public and secret key lengths are long, so these protocols are not suitable for implementation in environments with modest capability devices[9,10]. In this paper, we propose a modified protocol to improve upon Wang et al.'s to resist DoS and decrease communication cost.

The paper is organized as follows: In Section 2, we review Wang *et al.*'s scheme[7] and analyze its weaknesses. In Section 3, we propose our scheme. In Section 4, the security analysis of our proposed scheme is discussed in comparison with Wang *et al.*'s scheme. Finally, in Section 5, we conclude the paper.

## 2. Review and Analysis of the Wang et al.'s Scheme.

In 2011, Wang *et al.* proposed a secure and efficient identification and key agreement protocol with user anonymity (SEIKA)[7], they also proved security by using the random oracle model, and described other protocols[4,6,8,11] as not suitable in pervasive computing environments. A review and analysis of the Wang *et al.*'s scheme is given in this section.

### 2.1. The Wang *et al.*'s Scheme.

Wang *et al.*'s scheme[7] consists of two phases: a parameter generation phase, and an anonymous user identification and key agreement phase. Descriptions of these phases are given below.

Table 1. Notation

| | |
|---|---|
| $U_c$ | The identity of a client. |
| $U_s$ | The identity of a server. |
| $E_K(M)$ | A plaintext M encrypted using a secure symmetric key $K$. |
| $D_K(C)$ | A ciphertext C decrypted using a secure symmetric key $K$. |
| $t_1$ and $t_2$ | Two random numbers chosen by $U_c$ and $U_s$. |
| $H(\cdot)$ | Public one-way hash function and the output length is l. |

**Phase 1: Parameter generation phase:** The related parameters in this scheme are as follows:

1. A smart card producing center (SCPC) chooses a large prime number $q(q > 2^{160})$ and two field elements $(a, b)$. Where $a \in q, b \in q$ must satisfy$4a^3 + 27b^2 \bmod q \neq 0$, and the elliptic curve equation is defined as: $E_p$: $y^2 = x^3 + ax + b$.

2. The server generates a point $G$ from order $n$ which satisfies $n \times G = 0$ and $n/2^{160}$.

3. Every user $(U_c, U_s)$ has to register on the SCPC. For each user, the SCPC selects a random number $X_i$ and computes a public key $PK_i = X_i \times G$, where $X_i < n$.
4. A public key table which contains the identities and the public keys of the registered users in the SCPC.

**Phase 2: Anonymous user identification and key agreement phase:** User $i$ performs the following steps to log-in to the server:

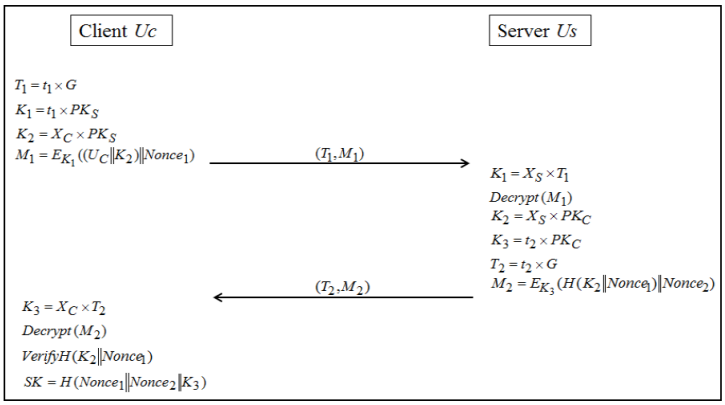


FIGURE 1. Anonymous user identification and key agreement phase

**Step 1:** The client selects a random number $t_1$, obtains the server's public key $PK_S = X_S \times G$ and calculates $K_1 = t_1 \times PK_S$ and $T_1 = t_1 \times G$. The client encrypts $M_1 = E_{K_1}(U_C \| Nonce_1)$ by using $K_1$. $T_1$ and $M_1$ are passed to the server.

**Step 2:** After the server receives $T_1$ and $M_1$, it will calculate $K_1 = X_S \times T_1$ and $D_{K_1}(M_1)$ and get $U_C$, $K_2$ and $Nonce_1$. The server will verify one thing:

1. Is $U_C$ stored in the public key table?
2. Is the value of $K_2$ correct? (This is to confirm whether someone forged a legitimate client identity.)

If either of these checks is false, the server revokes the agreement. If the checks are true, the server chooses a random number $t_2$ and retrieves the client's public key $PK_C = X_C \times G$ from public key table. The server calculates $K_2 = X_S \times PK_C$, $K_3 = t_2 \times PK_C$, and $T_2 = t_2 \times G$. Then, the server encrypts $M_2 = E_{K_3}(H(K_2 \| Nonce_1) \| Nonce_2)$ by using $K_3$. Finally, the server sends $(T_2, M_2)$ to the client.

**Step 3:** The client calculates $K_2 = X_C \times T_2$ and employs it to decrypt $M_2$ and to verify $H(K_3 \| Nonce_1)$ is correct or not. Then the client calculates $K_3 = X_C \times PK_S$ and sends $M_3 = H(K_3 \| Nonce_2)$ to the server and calculates the session key $SK = H(Nonce_1 \| Nonce_2 \| K_3)$.

**Step 4:** The sever checks $(K_3 \| Nonce_2)$. If they are the same, the server calculates a session key $SK = H(Nonce_1 \| Nonce_2 \| K_3)$.

2.2. **Security Analysis of Wang *et al.*'s Scheme.** An attacker can initiate a registration and by sending $T_1 \& M_1$ to the server. The server verifies the identity that actually is the attacker disguised, and calculates the next step. The attacker need not respond the $T_2 \& M_2$ that the server sends. So the server calculates $T_2 \& M_2$, and awaits message $M_3$. In this way, multiple incomplete handshakes can exhaust server resources.

3. **The Proposed Scheme.** We improve on Wang *et al.*'s scheme[7] and propose an improved identification and key agreement protocol. Our proposed scheme consists of two phases: a parameter generation phase, and an anonymous user identification and key agreement phase.

**Phase 1: Parameter generation phase:** This phase is modelled after Wang *et al.*'s scheme. The related parameters in this scheme are as follows:

1. A smart card producing center (SCPC) chooses a large prime number $q(q > 2^{160})$ and two field elements $(a, b)$. Where $a \in q, b \in q$ must satisfy$4a^3 + 27b^2 \bmod q \neq$

0, and the elliptic curve equation is defined as: $E_p$: $y^2 = x^3 + ax + b$.

2. The server generates a point $G$ from order $n$ which satisfies $n \times G = 0$ and $n/2^{160}$.

3. Every user $(U_c, U_s)$ has to register on the SCPC. For each user, the SCPC selects a random number $X_i$ and computes a public key $PK_i = X_i \times G$, where $X_i < n$.
4. A public key table which contains the identities and the public keys of the registered users in the SCPC.

**Phase 2: Anonymous user identification and key agreement phase:** User i performs the following steps to log-in to the server (as depicted in Figure 1):

**Step 1:** The client selects a random number $t_1$, obtains the server's public key $PK_S = X_S \times G$ and calculates $K_1 = t_1 \times PK_S$, $K_2 = X_C \times PK_S$ and $T_1 = t_1 \times G$. The client encrypts $M_1 = E_{K_1}(U_C \| K_2 \| Nonce_1)$ by using $K_1$.Then, $T_1$ and $M_1$ will be passed to the server.

**Step 2:** After the server receives $T_1$ and $M_1$, it will calculate $K_1 = X_S \times T_1$ and $D_{K_1}(M_1)$ and get $U_C$, $K_2$ and $Nonce_1$. The server will verify one thing:

1. Is $U_C$ stored in the public key table?
2. Is the value of $K_2$ correct? (This is to confirm whether someone forged a legitimate client identity.)

If either of these checks is false, the server revokes the agreement. If the checks are true, the server chooses a random number $t_2$ and retrieves the client's public key $PK_C = X_C \times G$ from public key table. The server calculates $K_2 = X_S \times PK_C$, $K_3 = t_2 \times PK_C$, and $T_2 = t_2 \times G$. Then, the server encrypts $M_2 = E_{K_3}(H(K_2 \| Nonce_1) \| Nonce_2)$ by using $K_3$. Finally, the server sends $(T_2, M_2)$ to the client.

**Step 3:** The client calculates $K_3 = X_C \times T_2$ and employs it to decrypt $M_2$ and to verify $H = (K_2 \| Nonce_1)$. Then the client calculates the session key $SK = H(Nonce_1 \| Nonce_2 \| K_3)$.

4. **Security Analysis and Comparison.** In this section, we will analyze the security of our proposed scheme and make comparisons with related schemes.

4.1. **Security Analysis.** We propose a modified protocol to improve Wang *et al.*'s scheme which resists DoS attacks and decreases communication cost. We discuss two different aspects of our approach:

**DoS attack resistance:** When the attacker initiates a registration, the method of Wang *et al.*[7] has to wait until the step three in order to authenticate the user. So the attacker can continue to send $T_1 \& M_1$ messages to the server. The server verifies the legitimate identity that is the attacker disguised, and calculates the next step. In our method, the attacker must produce $K_2$ by using the client's secret key $X_C$ and the server's public key $PK_S$, and send the message $M_1 = E_{K_1}(U_C \| K_2 \| Nonce_1)$ to the server in step one. Then the server calculates $K_2$ by using secret key $X_S$ and the client's public key $PK_C$ to check the value of $K_2$. When a message is forged without secret key $X_C$, the message will fail the check since $K_{2C} \neq K_{2S}$. This allows the server reject messages from attackers masquerading as legitimate clients from the beginning.

**Mutual authentication:** In step two, the server decrypts $M_1$ to get $K_2$, and verifies the client by using the value of $K_2$ to check if the client is not a forgery. The attacker does not know the value of $X_C$ in $K_2 = X_C \times PK_S$, so the server can confirm the legitimacy of the client by checking message $M_1$. In addition, the client can verify

the value of $H(K_2\|Nonce_1)$ in the server's $M_2$. If it is a masquerading server, when the client decrypts $M_2$ to verify, the interior value $H(K_2\|Nonce_1) \neq H(K_2\|Nonce_1^*)$ will not be the same. Using these two verifications, the client is no longer required to produce message M3. In our proposed scheme, the number of messages and cost of communication is less than Wang *et al.*'s method.

4.2. **Comparison.** The following table compares the properties of the proposed scheme and previous schemes:

TABLE 2. Properties of the proposed scheme versus previous schemes

| | Our scheme | Wang[7] | SIKA[6] | Lee[4] | Wu[8] | Yang[11] |
|---|---|---|---|---|---|---|
| C1 | Yes | Yes | Yes | Yes | Yes | Yes |
| C2 | Yes | Yes | Yes | No | No | No |
| C3 | Yes | Yes | Yes | Yes | Yes | Yes |
| C4 | Yes | Yes | No | No | No | No |
| C5 | Yes | Yes | No | No | No | No |
| C6 | Yes | Yes | No | No | No | No |
| C7 | Yes | Yes | Yes | No | Yes | Yes |
| C8 | Yes | Yes | Yes | Yes | Yes | No |
| C9 | Yes | Yes | Yes | No | Yes | Yes |
| C10 | Yes | Yes | No | Yes | No | Yes |
| C11 | Yes | No | Yes | Yes | Yes | No |
| C12 | Yes | Yes | No | No | No | No |

C1 no password table
C2 mutual authentication
C3 session key agreement
C4 low communication and computation cost
C5 no time-synchronization problem
C6 The participant does not need to hold the system or other participant's public key
C7 The identity of the client will be protection
C8 If a session key is compromised, it will not affect other session keys
C9 The simulation server cannot be deceived to the client
C10 No one can simulate a client to obtain services from the server
C11 The denial of service attack cannot work in the protocol
C12 Improve accuracy of the reliability and safety analysis

5. **Conclusion.** In this paper, we review Wang *et al.*'s scheme[7] and discuss the major drawbacks of their scheme. Then we proposed a modified protocol to improve Wang et al.'s scheme to resist DoS and also reduce computation and communication costs while maintain all the benefits of the Wang et al.'s scheme.

## REFERENCES

[1] J.W. Byun, D.H. Lee, J.I. Lim, EC2C-PAKA: An efficient client-to-client password authenticated key agreement, *Inform. Sci.*, vol. 117, no. 19, pp. 3995-4013, 2007.
[2] H.Y. Chien, Practical anonymous user authentication scheme with security proof, *Computers & Security*, vol. 27, no. 5-6, pp. 216-223, 2008.

[3] C.L. Hsu and Y.H. Chuang, A novel user identification scheme with key distribution preserving user anonymity for distributed computer networks, *Information Sciences*, vol.179, pp.422-429, 2009.
[4] W.B. Lee and C.C. Chang, User identification and key distribution maintaining anonymity for distributed computer network, *Comput. Syst. Sci. Engrg.*, vol. 15, pp.113-116, 2000.
[5] N.Y. Lee, C.N. Wu and C.C. Wang, Authenticated multiple key exchange protocols based on elliptic curves and bilinear pairings, *Computers & Electrical Engineering*, vol. 34, pp.12-20, 2008.
[6] K. Mangipudi and R. Katti, A secure identification and key agreement protocol with user anonymity (SIKA),*Computers & Security*, vol. 25, pp.420-425, 2006.
[7] R.C. Wang, W.S. Juang and C.L. Lei, Provably secure and efficient identification and key agreement protocol,*Journal of Computer and System Sciences*, vol.77, pp.790-798, 2011.
[8] T.S. Wu and C.L. Hsu, Efficient user identification protocol with key distribution preserving anonymity for distributed computer networks, *Computers & Security*, vol. 23, pp.120-125, 2004.
[9] T.Y. Wu and Y.M. Tseng, An efficient user authentication and key exchange protocol for mobile client-server environments, *Computer Networks*, vol. 54, no. 9, pp. 1520-1530, 2010.
[10] T.Y. Wu and Y.M. Tseng, An ID-based mutual authentication and key exchange protocol for low-power mobile devices, *The Computer Journal,* vol.53, no. 7, pp. 1062-1070, 2010.
[11] Y. Yang, S. Wang, F. Bao, J. Wang and D.H. Deng, New efficient user identification and key distribution protocol providing enhanced security, *Computers & Security*, vol. 23, pp.697-704, 2004.
[12] D. Nguyen, S. Oh, and B. You, A framework for Internet-based interaction of humans, robots, and responsive environments using agent technology, *IEEE Trans. Ind. Electron.*, vol.52, pp. 1521-1529, 2005.
[13] K. Saeed and M. Nammous, A speech-and-speaker identification system: Feature extraction, description, and classification of speech-signal Image, *IEEE Trans. Ind. Electron.*, vol.54, pp. 887-897, 2007.
[14] N. Koblitz, A. Menezes, and S. Vanstone, The state of elliptic curve cryptography, *Designs, Codes Cryptography*,vol. 19, pp. 173-193, 2000.
[15] W. Juang, Efficient password authenticated key agreement using smart cards, *Computer Security*, vol.23, pp. 167-173, 2004.