Journal of Computers Vol. 27, No. 4, 2016, pp. 85-93
doi:10.3966/199115592016122704007

An Improved Biometric-based Multi-server Authentication Scheme Using Smart Card

Jongho Moon[1], Jiseon Yu[2], Younsung Choi[1], and Dongho Won[1*]

[1] College of Information and Communication Engineering, Sungkyunkwan University, Suwon 16419, Korea
{jhmoon, yschoi, dhwon}@security.re.kr

[2] Center for Information Security Technologies (CIST), Korea University, Seoul 02841, Korea
gsun2@korea.ac.kr

Received 20 April 2015; Revised 29 September 2015; Accepted 6 October 2016

Abstract. Remote user authentication scheme is one of the most convenient authentication schemes to deal with secret data via insecure communication channel. During the last couple of decades, many researchers have proposed a remote user authentication schemes which are ID-based, password-based, and smart card-based. Above all, smart card-based remote user authentication schemes for multi-server environment are a widely used and researched method. One of the benefits of smart card-based authentication scheme is that a server does not have to keep a verifier table. Furthermore, remote user authentication scheme for multi-server environment has resolved the problem of users to manage the different identities and passwords. In 2015, Baruah et al. improved Mishra et al.'s scheme, and claimed that their scheme is more secure and practical remote user authentication scheme. However, we find that Baruah et al.'s scheme is still insecure. In this paper, we demonstrate that their scheme is vulnerable to outsider attack, smart card stolen attack, user impersonation attack and replay attack. To overcome these drawbacks, we propose a robust and more secure user authentication scheme. Finally, we show that our proposed scheme is more secure and supports security properties than Baruah et al.'s scheme.

Keywords: multi-server environment, remote user authentication, smart card

1 Introduction

Since Lamport [1] proposed the first password-based authentication scheme via insecure communication in 1981, password-based authentication schemes [2-8] have been extensively investigated. However, a problem of password-based authentication scheme is that a server must maintain a password table for verifying the legitimacy of a remote user. Therefore, the server requires additional memory space for storing the password table for verifying user identity. Furthermore, password is generally simple and can be easily broken or forgotten. For this reason, many researchers have proposed a new remote user authentication scheme by using biological characteristics of persons such as fingerprint, iris and so on. The main property of using biometric is its uniqueness. In the view of the fact that many remote user authentication schemes using biological characteristics [9-12] have been proposed. In 2010, Li and Hwang [13] proposed a remote user authentication scheme which was based on biometric verification, smart card, one-way hash function and nonce for authentication. However, in 2011, Li et al. [14] found that Li and Hwang's scheme does not provide proper authentication and cannot resist man-in-the-middle attack. After that, Chuang and Chen proposed an anonymous multi-server authentication scheme based on trust computing [15]. However, Mishra et al. demonstrated that the authentication scheme of Chuang and

† This paper is an extended version building on the results already presented in QSHINE 2015.
* Corresponding author

Chen cannot resist stolen smart card attack and impersonation attack and then proposed an improved multi-server based authentication scheme using smart cards for security enhancement [16]. In 2015, Baruah et al. [17] showed that the authentication scheme of Mishra et al. cannot withstand stolen smart card attack and impersonation attacks as well and proposed biometric-based remote user authentication scheme in multi-server environment. However, Baruah et al.'s authentication scheme is still insecure. We find that their scheme cannot withstand outsider attack, smart card stolen attack, impersonation attack and replay attack as well. To overcome these drawbacks, we propose a robust and more secure user authentication scheme.

2 Review in Baruah et al.'s Scheme

This section reviews the biometric-based multi-server authentication scheme proposed by Baruah et al. in 2015. As previous researches, Baruah et al.'s scheme consists of four phases: registration, login, authentication and password change phases which as follows. The notations used in this paper are summarized as Table 1.

Table 1. Notations used in Baruah et al.'s scheme

Notations	Description
TD_i	Identity of the i^{th} user
STD_j	Identity of the j^{th} server
PW_i	Password of the i^{th} user
BTO_i	Biometric of the i^{th} user
PSK	Pre-shared key of the servers
X	Master secret maintained by the registration center
$h(\cdot)$	A collision resistant one-way hash function
N_i, n_1	Random nonce of the i^{th} user
N_j, n_2	Random nonce of the j^{th} server
$\oplus$	The bitwise XOR operation
$\|\|$	Message concatenation operation

2.1 Registration phase

The registration phase is the initial phase of the scheme. In this phase, the registration center provides the secrets to the user as well as the server. It consists of the server registration phase and the user registration phase.

2.1.1 Server registration phase

When a server wants to provide some service to the public, then it has to first register itself to the registration center. The server sends a join request along with its identity (say, STD_j) to the registration center. In return, the registration center replies with $h(STD_j||h(PSK))$ and $h(PSK||X)$ through the Internet Key Exchange Protocol version 2 (IKEv2) [18]. The server uses these secret to authenticate any registered user.

2.1.2 User registration phase

The users must first register themselves if they want to access any services provided by the set of registered servers. Therefore, the user submits his/her identity TD_i and $R_1 = h(PW_i||BTO_i)$ via a secure channel. Then, the registration center performs the following.

(1) The registration center computes,

$A_i = h(TD_i||X)$

$B_i = h(PSK||X) \oplus A_i$

$C_i = h(R_1||TD_i) \oplus h(A_i)$

$D_i = h(PSK) \oplus h(TD_i)$

$\varepsilon_i = R_1 \oplus TD_i$

(2) Then, the registration center creates a smart card SC_i with the following information $\{B_i, C_i, D_i, \varepsilon_i, h(\cdot)\}$ and sends the smart card to user over a secure channel.

2.2 Login phase

To start any conversation, the user must first login to a specific terminal using smart card. The user inserts his/her the smart card into card-reader and inputs his/her identity TD_i, password PW_i and biometric information BTO_i. Then, the smart card executes the following sequence of operations.

(1) The smart card before sending any information to the server first checks whether the user is authorized to gain access or not. Therefore, it computes $R_1 = h(PW_i||BTO_i)$ and then verifies whether the entered identity TD_i is equal to stored identity $TD_i = R_1 \oplus \varepsilon_i$ or not. If failure occurs, the login phase is immediately aborted. Otherwise, proceeds for the succeeding steps.

(2) After checking the identity of user, the smart card extracts $h(PSK) = h(TD_i) \oplus D_i$ and $h(A_i) = C_i \oplus h(R_1||TD_i)$ from the stored data.

(3) Then, the smart card randomly generates a nonce N_i and computes the messages.

$M_1 = h(STD_j||h(PSK)) \oplus h(TD_i||N_i)$

$M_2 = N_i \oplus h(A_i)$

$v_1 = h(N_i \oplus B_i)$

(4) The smart card transmits the login request message $\{B_i, M_1, M_2, v_1\}$ to the server STD_j over a public channel for authentication.

2.3 Authentication phase

After receiving the login request messages, the server STD_j performs the following set of operations to agree on the same session key.

(1) The server uses its secrets, obtained during registration, to compute $A_i = B_i \oplus h(PSK||X)$ and $h(TD_i||N_i) = M_1 \oplus h(STD_j||h(PSK))$. Using $h(A_i)$, it gets N_i from $M_2 : N_i = M_2 \oplus h(A_i)$.

(2) Before generating any messages, the server must verify the user's authenticity. So, it uses the above derived information and verifies whether v_1 is equal to the computed value $h(N_i \oplus B_i)$ or not. If this holds, then the server generates a random nonce N_j. On failure, the phase is simply exited.

(3) The server uses the user's information and its nonce N_j and identity STD_j to generate the session key as $SK_{ji} = h(h(TD_i||N_i)||STD_j||B_i||N_j)$.

(4) Now, the server sends its randomly selected nonce to the user as $M_3 = N_j \oplus h(TD_i||N_i)$ and also $v_2 = N_i \oplus h(SK_{ji}||N_j)$ over a public channel.

(5) Once the message is received, the user computes N_j from M_3. It then uses the information to compute the session key as $SK_{ij} = h(h(TD_i||N_i)||STD_j||B_i||N_j)$. It is to be noted that both session keys are the same.

(6) Now, the user verifies whether the server is the actual one or not with whom he/she wants to communicate with. It is done by checking Ni with the computed value $v_2 \oplus h(SK_{ji}||N_j)$.

2.4 Password change phase

The mechanism is simple enough that if the user wants to change his/her password, it can be done without informing the registration center.

(1) The user inserts his/her smart card into card-reader and enters his/her identity TD_i, password PW_i and biometric BTO_i.

(2) Smart card checks the entered information. If the user is the authentic one, the smart card prompts the user for new password PW_i^* and computes,

$R_1^* = h(PW_i^*||BTO_i)$

$\varepsilon_i^* = \varepsilon_i \oplus R_1 \oplus R_1^*$

$C_i^* = h(R_1^*||TD_i) \oplus h(R_1||TD_i) \oplus C_i$

(3) Lastly, the smart card updates ε_i^* and C_i^* in the place of ε_i and C_i. Now, the updated smart card has $SC_i = \{B_i, C_i^*, D_i, \varepsilon_i^*, h(\cdot)\}$

3 Security Analysis of Baruah et al.'s Scheme

In this section, we demonstrate the vulnerability of Baruah et al.'s scheme in various communication scenarios. The following assumptions are made during the analysis and design of the scheme.

(1) An adversary can be either a user or a server. A registered user can act as an adversary.

(2) An adversary can eavesdrop every communication in public channels. He/she can capture any message exchanged between user and server.

(3) An adversary has the ability to alter, delete or reroute the captured message.

(4) Information can be extracted from the smart card by examining the power consumption of the card.

3.1 Outsider attack

Any adversary U_a who is the legal user and owns a smart card can obtain information $\{B_a, C_a, D_a, \varepsilon_a, h(\cdot)\}$ and then he/she can compute $h(PSK) = D_a \oplus h(TD_a)$. Thus, an adversary U_a can obtain $h(PSK)$ which same for each legal user and is the hash value of pre-shared key of the servers.

3.2 Smart card stolen & off-line identity guessing attack

Smart card stolen attack means an adversary who possessed with smart card performs any operation which the smart card and obtains any information. If an outsider adversary U_a steals the smart card of legitimate user U_i and obtains parameters $\{B_i, C_i, D_i, \varepsilon_i, h(\cdot)\}$, then he/she can easily compute out the hash value of the identity of the user U_i by computing $D_i \oplus h(PSK)$. Now, an adversary U_a performs an off-line identity guessing to get the current identity of the user U_i.

(1) The outsider adversary calculates $h(TD_i) = D_i \oplus h(PSK)$.

(2) Then, the adversary selects a random identity TD_i^*, calculates $h(TD_i^*)$ and compares it with $h(TD_i)$. If the result is equal to $h(TD_i)$, the adversary infers that TD_i^* is user U_i's identity. Otherwise the adversary selects another identity nominee and performs the same processes, until he locates the valid identity.

(3) After computing the identity of user U_i, an adversary can compute $R_1 = \varepsilon_i \oplus TD_i$ and $h(A_i) = C_i \oplus h(R_1||TD_i)$.

3.3 User Impersonation Attack

An outsider and smart card stolen adversary U_a can get the value $h(PSK)$ from his own card which is same for each user and the values TD_i, $h(A_i)$ from legitimate user U_i's smart card. Then, he/she can easily impersonate as user U_i to login and access the remote server because he can compute $\{B_i, M_1, M_2, v_1\}$.

(1) The adversary randomly generates a nonce N_i.

(2) Then, the adversary calculates,

$M_1 = h(STD_j||h(PSK)) \oplus h(TD_i||N_i)$

$M_2 = N \oplus {}_i h(A_i)$

$v_1 = h(N_i \oplus B_i)$

(3) After computing parameters, an adversary transmits the login request message $\{B_i, M_1, M_2, v_1\}$ to the server STD_j over a public channel for authentication.

3.4 Replay attack

An outsider adversary U_a eavesdrop a communication between a user and the server and then may try to use these messages for opening a communication to a server in future. An outsider adversary U_a may eavesdrop a communication and store the login request messages $\{B_i, M_1, M_2, v_1\}$, for performing replay attack in future where $M_1 = h(STD_j||h(PSK)) \oplus h(TD_i||N_i)$, $M_2 = N_i \oplus h(A_i)$, $v_1 = h(N_i \oplus B_i)$ and $B_i =$

$h(PSK||\ X) \oplus A_i$. Now, he/she can compute $h(TD_i||N_i) = M_1 \oplus h(STD_j||h(PSK))$. After computing $h(TD_i||N_i)$, the adversary transmits these stored messages $\{B_i, M_1, M_2, v_1\}$ to a registered server STD_j. The server STD_j, upon receiving the messages retrieves $A_i = B_i \oplus h(PSK||X)$, $h(TD_i||N_i) = M_1 \oplus h(STD_j||h(PSK))$, $N_i = M_2 \oplus h(A_i)$ and also verifies these using v_1. This verification holds, since the messages has not been modified by the adversary. Upon verification, the server STD_j selects a random nonce N_j and generates the session key as $SK_{ji} = h(h(TD_i||N_i)||STD_j||B_i||N_j)$. It then uses his/her session key for computing the reply messages $M_3 = N_j \oplus h(TD_i||N_i)$ and $v_2 = N_i \oplus h(SK_{ji}||N_j)$, and transmits to the adversary. Then, the outsider adversary easily can compute $N_j = M_3 \oplus h(TD_i||N_i)$ and $SK_{ij} = h(h(TD_i||N_i)||STD_j||B_i||N_j)$, because he/she knows $h(TD_i||N_i)$.

4 Our Proposed Scheme

In this section, we describe more secure remote user authentication. Our improved scheme consists of three phases: registration, login and authentication, and password changing phase. The notations used in our proposed scheme are summarized as Table 2.

Table 2. Notations used in our proposed scheme

Notations	Description
TD_i	Identity of the i^{th} user
STD_j	Identity of the j^{th} server
PW_i	Password of the i^{th} user
BTO_i	Biometric of the i^{th} user
RC	The registration center
PSK	Pre-shared key of the servers
X	Master secret maintained by the registration center
$h(\cdot)$	A collision resistant one-way hash function
y_i	A random number unique to user selected by RC
$\oplus$	The bitwise XOR operation
$\|\|$	Message concatenation operation

4.1 Architecture of our proposed scheme

In 2014, Binu et al. [19] proposed a remote user authentication scheme for multi-server environments using single sign on. Single sign on (SSO) [20] is an authentication mechanism that enables a user to sign-on once and access the services of various service providing (*SP*) servers in the same session. The security assertion mark-up language (*SAML*) [21] is a browser-based *SSO* protocol that allows a user to sign in only once at his/her identity provider (*TdP*). In this paper, we also describes a method of implementing two factor authentication using single sign on. The architecture of our proposed scheme includes four types of participants: a registration center (*RC*), authentication server (*AS*), *SP* servers and users. The *RC* and *AS* are trusted domain and provide the functionality of the identity provider (*TdP*).

4.2 Registration phase

The registration phase is the initial phase of the scheme. In this phase, the registration center provides the secrets to the user as well as the server. It consists of the server registration phase and the user registration phase.

4.2.1 Server registration phase

When a server wants to be a part of the multi-server environment (*MSE*), the server sends the registration request to the registration center *RC* along with server's identity STD_j. *RC* generates the pre-shared key *PSK* and the master secret key *X*, and computes $h(PSK||X)$. Then, *RC* shares the *PSK* and $h(PSK||X)$ with the *TdP*.

4.2.2 User registration phase

The user U_i who wants to become a registred member of the system selects his/her identity TD_i and password PW_i. Then, U_i computes $R_1 = h(PW_i||BTO_i)$ and sends the registration request messages $\{TD_i, R_1\}$ to registration center *RC* over a secure channel. After receiving the registration request message, the registration center *RC* performs the following.

(1) The registration center computes,

$A_i = h(TD_i||X)$

$B_i = h(PSK||X) \oplus A_i$

$C_i = y_i \oplus h(A_i)$

$D_i = h(y_i||PSK) \oplus h(R_1||TD_i)$

$\varepsilon_i = h(TD_i||R_1)$

(2) Then, the registration center *RC* creates a smart card SC_i with the following information $\{B_i, C_i, D_i, \varepsilon_i, h(\cdot)\}$ and sends the smart card to user over a secure channel.

4.3 Login and authentication phase

A user U_i attempts to access a protected resource on the *SP* over his/her browser. After receiving the request from the browser, the *SP* generates a *SAML* request for authenticating the client. This request includes STD_j of the *SP* and *assertion consumer service* URL of the *SP* to post the final *SAML* response. The *SP* redirects the request to the browser. In order to enhance security, we proposed a two-factor authentication scheme based on biometrics and smart card.

(1) The browser on getting the redirect, issues a *HTTPS GET* request containing the *SAML* request to the *TdP*.

(2) The *TdP* checks for a valid session with the browser. If there is no existing session between the browser and the *TdP*, then generates a session and authenticates the client using our proposed authentication scheme explained as follows. *TdP* presents the user with a login form where in the user needs to input the identity TD_i and password PW_i. Since the method of authentication is 2*FA* and we are using the smart card as the second factor, the user U_i needs to submit his/her smart card at the card reader. Hence, U_i imprints his/her biometric BTO_i. The smart card before sending any information to *TdP* first checks whether the user is authorized to gain access or not. Therefore, it computes $R_1 = h(PW_i||BTO_i)$ and then verifies whether the $h(TD_i||R_1)$ is equal to stored ε_i or not. If failure occurs, the smart card terminates the session. Otherwise, proceeds for the succeeding steps. After checking the identity of user, the smart card extracts $h(y_i||PSK) = h(R_1||TD_i) \oplus D_i$ from the stored data.

(3) Then, the smart card randomly generates a nonce N_i and computes the messages.

$M_1 = h(STD_j||h(y_i||PSK)) \oplus h(TD_i||N_i)$

$M_2 = N_i \oplus h(y_i||PSK)$

$v_1 = h(N_i \oplus B_i)$

(4) The smart card transmits the login request message $\{B_i, C_i, M_1, M_2, v_1\}$ to *TdP* over a secure channel for authentication.

(5) After receiving the login request messages, *TdP* computes $A_i = B_i \oplus h(PSK||X)$, $y_i = C_i \oplus h(A_i)$ and $h(TD_i||N_i) = M_1 \oplus h(STD_j||h(y_i||PSK))$. Using $h(y_i||PSK)$, it gets N_i from $M_2 : N_i = M_2 \oplus h(y_i||\ PSK)$

(6) Before generating any messages, *TdP* must verify the user's authenticity. So, it uses the above derived information and verifies whether v_1 is equal to the computed value $h(N_i \oplus B_i)$ or not. If this holds, then *TdP* authenticates U_i.

(7) The *TdP* generates a *SAML* response which includes a *SAML* assertion containing the authentication status. The assertion is digitally signed by *TdP* and *TdP* sets its cookie here. Then, *TdP* redirects the *SAML* response to the browser.

(8) After checking the response status, the browser redirects the response to *SP* who provides access to resources.

4.5 Password change phase

The mechanism is simple enough that if the user wants to change his/her password, it can be done without informing the registration center.

(1) The user inserts his/her smart card into card-reader and enters his/her identity TD_i, password PW_i and biometric BTO_i.

(2) Smart card checks the entered information. If the user is the authentic one, the smart card prompts the user for new password PW_i^* and computes,

$R_1^* = h(PW_i^* || BTO_i)$

$D_i^* = h(y_i || PSK) \oplus h(R_1^* || TD_i)$

$\varepsilon_i^* = h(TD_i || R_1^*)$

(3) Lastly, the smart card updates D_i^*, ε_i^* in the place of D_i, ε_i. Now, the updated smart card has $SC_i = \{B_i, C_i, D_i^*, \varepsilon_i^*, h(\cdot)\}$

5 Security Analysis of Our Proposed Scheme

In this section, we demonstrate that our scheme can withstand several possible attacks. We also show that our scheme supports several security properties.

5.1 Resisting outsider attack

Suppose an outsider adversary U_a extracts all information $\{B_a, C_a, D_a, \varepsilon_a, h(\cdot)\}$ from own smart card by side channel attack [22]. However, he/she cannot obtain any secret information. U_a can compute $h(y_a||PSK) = h(R_a||TD_a) \oplus D_a$. However, the value y_a is a random number unique to user selected by *RC*, and *PSK* is the pre-shared secret key between *RC* and *AS*. Therefore, U_a does not know any secret information and our proposed scheme can resist outsider attack.

5.2 Resisting stolen smart card attack

If an adversary U_a steals U_i's smart card, then U_a can extract security parameters $\{B_i, C_i, D_i, \varepsilon_i, h(\cdot)\}$ from legitimate user U_i's smart card. However, this information does not help them. He/She cannot obtain any information of U_i's TD_i and PW_i because is protected by secret parameters. Therefore, our proposed scheme can resist stolen smart card attack.

5.3 Resisting user impersonation attack

In the our proposed scheme, only U_i can compute $h(y_i||PSK) = h(R_1||TD_i) \oplus D_i$ since only he/she has the secrets BTO_i and password PW_i and *TdP* can compute $A_i = B_i \oplus h(PSK||X)$ since only he/she has the secrets $h(PSK||X)$. The *TdP* authenticates U_i by checking $h(N_i \oplus B_i) =? v_1$. Thus, our proposed scheme can resist user impersonation attack.

5.4 Resisting replay attack

Our proposed scheme uses the random values to resist replay attack. Each random value is generated for each session. Even if an adversary tries a replay attack, an adversary does not know this random value. Thus, our proposed scheme can resist replay attack.

5.5 Comparison security properties

We compare the proposed scheme with the other schemes regarding resistance to possible attacks as depicted by Table 3. Our proposed scheme resists all those attacks to which the previous schemes.

Table 3. Comparison of security attributes

Security attributes	Our	[13]	[15]	[16]	[17]
User anonymity	Yes	No	Yes	Yes	No
Biometric template	Yes	Yes	Yes	Yes	Yes
Simple password change	Yes	Yes	Yes	Yes	Yes
Mutual authentication	Yes	No	Yes	Yes	Yes
Resist impersonation attack	Yes	No	No	No	No
Resist stolen smart card attack	Yes	No	No	No	No
Resist offline guessing attack	Yes	Yes	Yes	Yes	Yes
Resist man-in-the-middle attack	Yes	No	No	Yes	Yes
Resist insider attack	Yes	No	Yes	Yes	Yes
Time synchronization	No	No	No	No	No

6 Conclusion

In 2015, Baruah et al. proposed an enhanced scheme of Mishra et al.'s scheme and demonstrated it is resistance to famous attacks such as impersonation attacks, smart card stolen attacks, off-line password guessing attacks, man-in-the-middle attacks and replay attacks. However, Baruah et al.'s scheme is still insecure. We show that their scheme can suffer from outsider attack, smart card stolen attack, user impersonation attack and replay attack. In this paper, to solve the security vulnerabilities, we proposed an improved protocol for authentication scheme that keeps the similar properties of their scheme and make it more secure. The security analysis explains that our proposed scheme rectifies the weaknesses of Baruah et al.'s scheme.

Acknowledgement

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.R0126-15-1111, The Development of Risk-based Authentication· Access Control Platform and Compliance Technique for Cloud Security)

References

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

[2] A. Conklin, G. Dietrich, D. Walz, Password-based authentication: a system perspective, System Sciences 50(2004) 629-631.

[3] M. Abdalla, P. Fouque, D. Pointcheval, Password-based authenticated key exchange in the three-party setting, On Public Key Cryptography-PKC 2005 3386(2005) 65-84.

[4] S. Jiang, G. Gong, Password based key exchange with mutual authentication, Selected Areas in Cryptography 3357(2005) 267-279.

[5] R. Gennaro, Y. Lindell, A framework for password-based authenticated key exchange, ACM Transactions on Information and System Security (TISSEC) 9(2006) 181-234.

[6] Y. Yang, R. Deng, F. Bao, A practical password-based two-server authentication and key exchange system, Dependable and Secure Computing 3(2006) 105-114.

[7] A. Groce, J. Katz, A new framework for efficient password-based authenticated key exchange, in Proc. of the 17th ACM Conference on Computer and Communications Security, 2010.

[8] I. Jeun, M. Kim, D. Won, Enhanced password-based user authentication using smart phone, Advances in Grid and Pervasive Computing 7296(2012) 350-360.

[9] J. Lee, S. Ryu, K. Yoo, Fingerprint-based remote user authentication scheme using smart cards, Electronics Letters 38(2002)

554-555.

[10] C. Lin, Y. Lai, A flexible biometrics remote user authentication scheme, Computer Standards & Interfaces 27(2004) 19-23.

[11] C. Chang, I. Lin, Remarks on fingerprint-based remote user authentication scheme using smart cards, ACM SIGOPS Operating Systems Review 38(2004) 91-96.

[12] W. Yi, S. Kim, D. Won, Smart card based AKE protocol using biometric information in pervasive computing environments, in: Proc. Computational Science and Its Applications-ICCSA 2009, 2009.

[13] C. Li, M. Hwang, An efficient biometrics-based remote user authentication scheme using smart cards, Journal of Network and Computer Applications 33(2010) 1-5.

[14] X. Li, J. Niu, J. Ma, W. Wang, C. Liu, Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards, Journal of Network and Computer Applications 34(2011) 73-79.

[15] M. Chuang, M. Chen, An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics, Experts Systems with Applications 41(2014) 1411-1418.

[16] D. Mishra, A. Das, S. Mukhopadhyay, A secure user anonymity preserving biometric-based multi-server authenticated key agreement scheme using smart cards, Expert Systems with Applications 41(201) 8129-8143.

[17] K. Baruah, S. Banerjee, M. Dutta, C. Bhunia, An improved biometric-based multi-server authentication scheme using smart card, International Journal of Security and Its Applications 9(2015) 397-408.

[18] C. Kaufman, Internet Key Exchange(IKEv2) protocol, 2005.

[19] S. Binu, M. Misbahuddin, P. Raj, A single sign based on secure remote user authentication scheme for multi-server environments, in: Proc. of International conference on Computer and Communications Technologies (ICCCT), 2014.

[20] OASIS, Security assertion mark up language, V2.0, technical overview. 〈http://docs.Oasis-open.org/Security/Saml/Post2.0/sstc-samltech-overview-2.0-cd-02.html〉.

[21] G. Thomas, Security analysis of the SAML single-sign-on browser/artifact profile. 〈https://www.acsac.org/2003/papers/73.pdf〉.

[21] T. Messerges, E. Dabbish, R. Sloan, A flexible biometrics remote user authentication scheme, IEEE Transactions on Computers 51(2002) 541-552.