Improved Secure Remote User Authentication Protocol

Ji-Seon Lee*, Ji Hye Park**, Jik Hyun Chang***° *Regular Members*

ABSTRACT

Recently, Hölbl *et al.* proposed an improvement to Peyravian-Jeffries's password-based authentication protocol to overcome some security flaws. However, Munilla *et al.* showed that Hölbl *et al.*'s improvement is still vulnerable to off-line password guessing attack. In this paper, we provide a secure password-based authentication protocol which gets rid of the security flaws of Hölbl *et al.*'s protocol.

Key Words : Security, Access Control, Authentication, Dictionary Attack, Denial-Of-Service Attack

Ⅰ. Introduction

A password-based user authentication method is widely used to provide a (remote) access control to network applications. In order to identify himself/herself and request services, a user typically provides his/her identity(ID) and password to a client application program, and the client program on behalf of the user interacts with a server. Usually the server maintains a database including identification information corresponding to the user and services needed to be authorized. A password-based authentication method has many merits in views of mobility and efficiency because of the use of only human-memorable password for authentication.

However, it is not easy to design a sufficiently secure password-based authentication protocol. The main difficulty of constructing a secure password-based authentication protocol stems from an intrinsic structure of the protocol treating a low-entropy password. For a human to memorize a password easily, a password might have low-entropy i.e., 4 or 8 characters comprised of a natural language phrase drawn from a relatively small dictionary. A password-based authentication protocol is easily susceptible to an off-line dictionary attack, also known as a password guessing attack, when transcripts generated in the protocol leak some meaningful information. The off-line dictionary attack is that an adversary uses the information to find a valid password by exhausting all possible candidate passwords in a dictionary. A password-based authentication protocol should be strong enough to resist an off-line dictionary attack. In addition, a password-based authentication protocol provides a reliable and efficient password refreshment method to periodically update a low-entropy password. This method enhances security of the protocol by minimizing a security loss from unexpected exposure of a password.

Since the initial work[1], various protocols have been suggested to achieve such security properties while maintaining good efficiency. Among them, Peyravian and Jeffries[6] proposed several protocols to provide remote user authentication using a password without any encryption algorithms. Subsequently, Munilla and Peinado[4] presented an off-line dictionary attack on Peyravian and Jeffries protocol and Shim[7] independently presented off-line dictionary and Denial-of-Service(DoS) attacks on the protocol. Recently, Hölbl *et al.*[3] presented an improved variant of (Diffie-Hellman based[2]) Peyravian and Jeffries protocol to resist the

* 고려대학교, 정보경영공학전문대학원, BK21 유비쿼터스 정보보호사업단 (jslee702@korea.ac.kr),
** 티맥스소프트 (toytoy1000@sogang.ac.kr), ***서강대학교 컴퓨터공학과 (jchang@sogang.ac.kr) (° : 교신저자)
논문번호 : KICS2009-03-118, 접수일자 : 2009년 3월 17일, 최종논문접수일자 : 2009년 9월 3일

attacks presented by Munilla and Peinado, and Shim. Hölbl *et al.*'s protocol consists of two sub-protocols, remote user authentication and password change protocols. Unfortunately, very recently, Munilla and Peinado showed that Hölbl *et al.*'s improved protocol also suffers from off-line password guessing attack[5].

In[5], Munilla and Peinado showed that the security flaw of Hölbl *et al.* protocol is leaking a common secret key shared between a client and a server. That is, anyone can easily compute the key from public transcripts generated for an execution of the protocol. In this paper, we discuss that such security flaw causes Denial-of-Service attacks due to its insecure structure. Finally, we present an improved remote user authentication scheme based on Hölbl *et al.* protocol.

The rest of this paper is organized as follows. In Section 2, we briefly review Hölbl *et al.*'s user authentication and password change protocols. In Section 3, we show that the protocols are vulnerable to both off-line password guessing attack and DoS attack. In Section 4, we provide a secure password-based authentication scheme and give security analysis of the proposed scheme in section 5. Concluding remarks are given in Section 6.

Ⅱ. Review of Hölbl *et al.*'s Protocol

In this section we briefly review Hölbl *et al.*'s protocol[3]. The protocol consists of two sub-protocols, user authentication and user password change protocols. The password change protocol consists of mutual authentication and password update phases. For more details, refer to[3]. Throughout the paper, we use the following notations.

- C, S : A client and a server, respectively
- E : An adversary
- ID, PW : User U's identity and password
- p : A large prime number
- $GF(p)$: The set of integers $\{0, 1, 2, \ldots, p-1\}$ with arithmetic operations defined modulo p
- g : A (multiplicative) primitive element over $GF(p)$
- H : A collision resistant one-way hash function
- $IDPW_{dig}$: $IDPW_{dig} = H(ID, PW)$ is a password digest value stored in server's database
- $\oplus$: Bitwise exclusive or operation

2.1 The user authentication protocol

The user authentication protocol is described as follows: We first assume that a user U has (or may memorize) his/her (ID, PW) and a server S stores $IDPW_{dig}$ instead of the password PW itself.

(1) The user U submits his/her ID and PW to the client C. Then C generates a random value r_C, chooses a large prime p and a primitive element $g \in GF(p)$. C chooses a large random integer $x < p-1$, computes $g^x \bmod p$, a password digest $IDPW_{dig}$ and m_x as follows:

$$IDPW_{dig} = H(ID, PW),$$
$$m_x = g^x \oplus H(ID, IDPW_{dig}).$$

Then it sends message $\{ID, r_C, p, g, m_x\}$ to S.

(2) After receiving the message, S chooses a random value r_S, a large random integer $y < p-1$ and computes $g^y \bmod p$. S retrieves $g^x = m_x \oplus H(ID, IDPW_{dig})$ and computes a Diffie-Hellman (DH) key $g^{xy} \bmod p$. Next, S generates two one-time challenge tokens

$$ch_1 = r_S \oplus H(g^{xy}, IDPW_{dig}, r_C),$$
$$ch_2 = g^{xy} \oplus H(g^{xy}, IDPW_{dig}, r_C), \text{ and}$$
$$m_y = g^y \oplus H(ID, IDPW_{dig}).$$

Then S sends $\{ch_1, ch_2, m_y\}$ to C.

(3) On the receipt of the message, C derives $g^y = m_y \oplus H(ID, IDPW_{dig})$ using $IDPW_{dig}$. Next, C computes $g^{xy} \bmod p$ and $h' = g^{xy} \oplus ch_2$. C then checks if $H(g^{xy}, IDPW_{dig}, r_C)$ is equal to h'. If the check fails, then C terminates the protocol. If the check succeeds,

C retrieves $r_S' = ch_1 \oplus h'$ and sends $\{ID, r_S'\}$ to S.

(4) After receiving $\{ID, r_S'\}$, S checks if the received r_S' is same as the generated r_S. If they are same, the user U is authenticated. Next, S generates a one-time authentication token sat and sends it to C.

$$sat = H(g^{xy}, IDPW_{dig}, r_C r_S)$$

(5) On the receipt of a server's authentication token, C verifies the validity of the received authentication token by independently computing $sat' = H(g^{xy}, IDPW_{dig}, r_C r_S')$ and comparing it with the received sat. If the server's authentication token is valid, S is authenticated.

(6) Both C and S may optionally establish a common session key to encrypt further information exchanged after this initial execution.

2.2 The password change protocol

The password change protocol is described as follows: The password change protocol consists of mutual authentication and password update phases. Because the mutual authentication phase is the same as step (1)-(4) in the user authentication protocol, we omit the mutual authentication phase and only describe the password update phase, that is, step (5) and (6).

(5) On the receipt of a server's authentication token, C verifies the validity of the received authentication token by independently computing $sat' = H(g^{xy}, IDPW_{dig}, r_C, r_S')$ and comparing it with the received sat. If they are the same, C generates a new password digest value $IDPW_{dig}'$ with a newly chosen password PW_{new}. Next, C generates one-time $mask$, mac, and m_IDPW_{dig} values as follows:

$$mask = H(g^{xy}, r_C, r_S),$$
$$mac = H(g^{xy}, IDPW_{dig}', r_C, r_S),$$
$$m\_IDPW_{dig} = mask \oplus IDPW_{dig}'.$$

Then C sends $\{ID, m\_IDPW_{dig}, mac\}$ to S.

(6) After receiving a message $\{ID, m\_IDPW_{dig}, mac\}$, S verifies the validity of the received mac. To do this, S computes

$$mask = H(g^{xy}, r_C r_S),$$
$$IDPW_{dig}' = mask \oplus m\_IDPW_{dig},$$
$$mac' = H(g^{xy}, IDPW_{dig}', r_C r_S).$$

And S compares it with the received mac. If it is valid, S sends a message to C accepting the password change. Also, S replaces $IDPW_{dig}$ with the new password digest value $IDPW_{dig}'$. Otherwise, it sends a message rejecting the password change. The password accept or reject message sent from S to C contains a protected response called $code$, where $flag$ is set to either 'accept' or 'reject' depending upon whether the password change is accepted or rejected.

$$code = H(g^{xy}, IDPW_{dig}, flag, r_C, r_S)$$

Ⅲ. Security Analysis of Hölbl *et al.*'s Protocol

Munilla and Peinado showed that Hölbl *et al.*'s improved protocol also suffers from off-line password guessing attack[5]. In their attack scenario, an adversary E eavesdrops all the transcripts honestly generated in an execution of Hölbl *et al.*'s protocol between a server S and a client C. That is, the adversary E captures transcripts, $\{ID, r_C p, g, m_x\}$, $\{ch_1, ch_2, m_y\}$, $\{ID, r_S'\}$, sat, $\{ID, m\_IDPW_{dig}, mac\}$, and $code$.

Using these eavesdropped transcripts, the adversary E can compute a DH key value as follows:

$$\begin{aligned} & r_S \oplus ch_1 \oplus ch_2 \\ &= r_S \oplus (r_S \oplus H(g^{xy}, IDPW_{dig}, r_C)) \\ &\quad \oplus (g^{xy} \oplus H(g^{xy}, IDPW_{dig}, r_C)) \\ &= g^{xy} \end{aligned}$$

The adversary with the DH key value g^{xy} can mount an off-line dictionary attack to find a user's current password as follows.

The adversary E with g^{xy} executes an off-line dictionary attack to find current password PW by

Client C		Server S
	[Mutual Authentication Phase]	
$r_C, p, g \in GF(p)$		
$x < p-1$		
$m_x = g^x \oplus H(ID, IDPW_{dig})$	$\xrightarrow{\{ID,\ r_C,\ p,\ g,\ m_x\}}$	$r_S, y < p-1$
		$m_y = g^y \oplus H(ID, IDPW_{dig})$
		$ch_1 = r_S \oplus H(g^{xy}, IDPW_{dig}, r_C)$
$g^y = m_y \oplus H(ID, IDPW_{dig})$	$\xleftarrow{\{ch_1,\ ch_2,\ m_y\}}$	$ch_2 = g^{xy} \oplus H(g^{xy}, IDPW_{dig}, r_C)$
$h' = ch_2 \oplus g^{xy}$		
if $h' \neq H(g^{xy}, IDPW_{dig}, r_C)$, then reject S		
else $r'_S = ch_1 \oplus h'$	$\xrightarrow{\{ID,\ r'_S\}}$	if $r'_S \neq r_S$, then reject C
if $sat \stackrel{?}{=} H(g^{xy}, IDPW_{dig}, r_C, r'_S)$, then accept S	$\xleftarrow{\{sat\}}$	else $sat = H(g^{xy}, IDPW_{dig}, r_C, r_S)$
	[Password Update Phase]	
select new password PW_{new}		
$IDPW'_{dig} = H(ID, PW_{new})$		
$mask = H(g^{xy}, r_C, r_S)$		
$mac = H(g^{xy}, IDPW'_{dig}, r_C, r_S)$		
$m\_IDPW_{dig} = mask \oplus IDPW'_{dig}$	$\xrightarrow{\{ID,\ m\_IDPW_{dig},\ mac\}}$	$mask = H(g^{xy}, r_C, r_S)$
		$IDPW'_{dig} = mask \oplus m\_IDPW_{dig}$
		$mac' = H(g^{xy}, IDPW'_{dig}, r_C, r_S)$
		if $mac' \stackrel{?}{=} mac$, then $IDPW_{dig} = IDPW'_{dig}$
		$code = H(g^{xy}, IDPW_{dig}, flag, r_C, r_S)$, where $flag =$ 'accept' or 'reject'
	$\xleftarrow{\{code\}}$	

Fig. 1. Hölbl *et al.*'s mutual authentication and password change protocol

iterating the following procedure upon all possible choices of PW':

(1) Pick a candidate password PW'.
(2) Compute $IDPW_{dig}{}' = H(ID, PW')$.
(3) Check if the following equalities hold:

$ch_1 = r_S \oplus H(g^{xy}, IDPW_{dig}{}', r_C)$ and
$ch_2 = g^{xy} \oplus H(g^{xy}, IDPW_{dig}{}', r_C)$.

Munilla and Peinado only discussed off-line dictionary attack to find current password. We found out that due to the above security flaw, using g^{xy}, r_C, r_S, and mac, the adversary is also able to mount an off-line dictionary attack even to find a new user's password. Similar to the previous off-line dictionary attack, the adversary E iterates the following procedure upon all possible choices of PW'.

(1) Pick a candidate password PW'.
(2) Compute $IDPW_{dig}{}' = H(ID, PW)$ and $mac' = H(g^{xy}, IDPW_{dig}{}', r_C, r_S)$.
(3) Check if $mac = mac'$.

Note that a match in the last step indicates a correct guess of the password with high probability due to pseudo-randomness of the one-way hash function H. Therefore, the adversary highly succeeds in guessing the valid password PW.

Furthermore, contrast to the previous off-line dictionary attack requiring relatively heavy computation overhead, an adversary can mount a Denial-of-Service attack easily. In password-based authentication, Denial-of-Service (DoS) attacks cause permanent error on authentication by introducing erroneous data during the procedures of

authentication. In Hölbl *et al.*'s protocol, once the adversary succeeds off-line password guessing attack, the adversary can update user's password of its choice using the password change protocol. Since then, the legal user cannot access the remote server and Denial-of-Service attack is succeeded.

Ⅳ. Improved Scheme

In this section we propose a new remote user authentication scheme. As previous schemes, our scheme is also composed of user authentication protocol and password change protocol. The security weakness of Hölbl *et al.*'s protocol is due to the insecure structure of the protocol. That is, the DH key value can be computed only with the intercepted messages between the server and the client. This is because three components (ch_1 and ch_2 in step (2) and $r_S'(r_S)$ in step (3)) of public transcripts leak common secret key g^{xy}. Therefore, we propose a new scheme that there is no way to get any meaningful information from the transmitted messages between the server and the client.

Our proposed scheme is not only secure against off-line password guessing attack but also secure against active attacks such as impersonation attack.

4.1 The user authentication protocol

As in Hölbl *et al.*'s protocol, we assume that a user U has (or may memorize) his/her (ID, PW) and a server S stores $IDPW_{dig}$ instead of the password PW itself.

(1) The user U submits his/her ID and PW to the client C. Then C generates a random value r_C, chooses a large prime p and a primitive element $g \in GF(p)$. C chooses a large random integer $x < p-1$, computes $g^x \bmod p$. Then it sends message $\{ID, r_C, p, g, g^x\}$ to S.

(2) After receiving the message, S chooses a random value r_S, a large random integer $y < p-1$ and computes $g^y \bmod p$. S masks g^y by computing y_1 as follows.

$$y_1 = g^y \oplus H(ID, IDPW_{dig})$$

And S also computes g^{xy}. The server generates two one-time challenge tokens

$$ch_1 = r_S \oplus g^{xy} \text{ and}$$

$$ch_2 = r_S \oplus H(g^{xy}, IDPW_{dig}, r_C).$$

S then sends $\{ch_1, ch_2, y_1\}$ to C.

(3) On the receipt of the message, C first computes $IDPW_{dig} = H(ID, PW)$ and recovers $g^y = y_1 \oplus H(ID, IDPW_{dig})$. Next, C computes g^{xy} and retrieves r_S' from the received token ch_1 by $r_S' = ch_1 \oplus g^{xy}$, and verifies the validity of the server with $ch_2 = r_S' \oplus H(g^{xy}, IDPW_{dig}, r_C)$. If it is not equal, the server is not genuine and the client terminates the protocol. If the equality holds, the client sends $\{ID, h(r_S')\}$ to S. Here, $h(\bullet)$ is a collision resistant one-way hash function.

(4) After receiving $\{ID, h(r_S')\}$, S checks if the received $h(r_S')$ is the same as the one it generated. If they are same, U is authenticated. Next, S generates a one-time authentication token sat and sends it to C.

$$sat = H(g^{xy}, IDPW_{dig}, r_C, r_S)$$

(5) On the receipt of a server's authentication token, C verifies the validity of the received authentication token by independently computing sat' and compares it with the received sat.

$$sat' = H(g^{xy}, IDPW_{dig}, r_C, r_S')$$

If the server's authentication token is valid, S is authenticated.

(6) Both C and S may optionally establish a common session key to encrypt further information exchanged after this initial execution.

4.2 The password change protocol

The password change protocol consists of

mutual authentication and password update phases. Because the mutual authentication phase is the same as step (1)-(4) in the user authentication protocol, we omit the mutual authentication phase and only describe the password update phase, that is, step (5) and (6).

(5) On the receipt of a server's authentication token, C verifies the validity of the received authentication token by independently computing sat' and comparing it with the received sat.

$$sat' = H(g^{xy}, IDPW_{dig}, r_C, r_S')$$

If they are the same, C generates a new password digest value $IDPW_{dig}'$ with a newly chosen password PW_{new}. Next, C generates one-time $mask$, mac, and m_IDPW_{dig} values as follows:

$$IDPW_{dig}' = H(ID, PW_{new}),$$
$$mask = H(g^{xy}, r_C, r_S),$$
$$mac = H(g^{xy}, IDPW_{dig}', r_C, r_S),$$
$$m\_IDPW_{dig} = mask \oplus IDPW_{dig}'.$$

Then C sends $\{ID, m\_IDPW_{dig}, mac\}$ to S.

(6) After receiving ID, m_IDPW_{dig}, and mac, S verifies the validity of the received mac. To do this, S computes

$$mask = H(g^{xy}, r_C, r_S),$$
$$IDPW_{dig}' = mask \oplus m\_IDPW_{dig},$$
$$mac' = H(g^{xy}, IDPW_{dig}', r_C, r_S).$$

And S compares it with the received mac. If it is valid, S sends a message to C accepting the password change. Also, S replaces $IDPW_{dig}$ with the new password digest value $IDPW_{dig}'$. Otherwise, it sends a message rejecting the password change. The password accept or reject message sent from S to C contains a protected response called $code$, where $flag$ is set to either 'accept' or 'reject' depending upon whether the password change is accepted or rejected.

$$code = H(g^{xy}, IDPW_{dig}, flag, r_C, r_S)$$

Ⅴ. Security Analysis of the Improved Scheme

In this section, we show that our improved protocol is secure against off-line password guessing attack, Denial-of-Service attack and active attack such as impersonation attacks.

5.1 Security against off-line password guessing attack

In the proposed scheme, to succeed off-line password guessing attack, the adversary should get g^{xy}. That is, for trying off-line password guessing attack, the adversary would use one of the followings.

$$ch_1 = r_S \oplus g^{xy}$$
$$ch_2 = r_S \oplus H(g^{xy}, IDPW_{dig}, r_C)$$

However, the client sends the hashed value of r_S instead of raw data to prevent off-line password guessing attack. Therefore, even if the adversary intercepts all the messages $\{ID, r_C, p, g, g^x\}$, $\{y_1, ch_1, ch_2\}$, and $\{ID, h(r_S)\}$, he cannot compute g^{xy} without knowing the value of r_S.

Since off-line password guessing attack cannot be performed without g^{xy} in the proposed scheme, it is infeasible to perform off-line password guessing attack.

5.2 Security against impersonation attacks

First, we consider the scenario of impersonation of server. Suppose that an adversary tries to impersonate the server. The adversary would intercept the message $\{ID, r_C, p, g, g^x\}$ and select random values y' and try to compute ch_1', ch_2' without knowing $IDPW_{dig}$. The adversary then sends $\{y_1', ch_1', ch_2'\}$ to the client. Upon receiving $\{y_1', ch_1', ch_2'\}$, the client recovers $g^{y'} = y_1' \oplus H(ID, IDPW_{dig})$ and computes $r_S' = ch_1' \oplus g^{xy'}$. Next, the client verifies that ch_2' equals to $r_S' \oplus h(g^{xy'}, IDPW_{dig}, r_C)$. However, since the adversary does not know the value $IDPW_{dig}$, it is

Client C		Server S
	[Mutual Authentication Phase]	
$r_C, p, g \in GF(p)$		
$x < p-1$		
	$\xrightarrow{\{ID,\ r_C,\ p,\ g,\ g^x\}}$	$r_S, y < p-1$
		$y_1 = g^y \oplus H(ID, IDPW_{dig})$
		$ch_1 = r_S \oplus g^{xy}$
$IDPW_{dig} = H(ID, PW)$	$\xleftarrow{\{ch_1,\ ch_2,\ y_1\}}$	$ch_2 = r_S \oplus H(g^{xy}, IDPW_{dig}, r_C)$
$g^y = y_1 \oplus H(ID, IDPW_{dig})$		
$r'_S = ch_1 \oplus g^{xy}$		
if $ch_2 \neq r'_S \oplus H(g^{xy}, IDPW_{dig}, r_C)$, then reject S		
else compute $h(r'_S)$	$\xrightarrow{\{ID,\ h(r'_S)\}}$	if $h(r'_S) \neq h(r_S)$, then reject C
if $sat \stackrel{?}{=} H(g^{xy}, IDPW_{dig}, r_C, r'_S)$, then accept S	$\xleftarrow{\{sat\}}$	else $sat = H(g^{xy}, IDPW_{dig}, r_C, r_S)$

	[Password Update Phase]	
select new password PW_{new}		
$IDPW'_{dig} = H(ID, PW_{new})$		
$mask = H(g^{xy}, r_C, r_S)$		
$mac = H(g^{xy}, IDPW'_{dig}, r_C, r_S)$		
$m\_IDPW_{dig} = mask \oplus IDPW'_{dig}$	$\xrightarrow{\{ID,\ m\_IDPW_{dig},\ mac\}}$	$mask = H(g^{xy}, r_C, r_S)$
		$IDPW'_{dig} = mask \oplus m\_IDPW_{dig}$
		$mac' = H(g^{xy}, IDPW'_{dig}, r_C, r_S)$
		if $mac' \stackrel{?}{=} mac$, then $IDPW_{dig} = IDPW'_{dig}$
		$code = H(g^{xy}, IDPW_{dig}, flag, r_C, r_S)$,
	$\xleftarrow{\{code\}}$	where $flag$ = 'accept' or 'reject'

Fig. 2. Proposed mutual authentication and password change protocol

impossible for the adversary to compute $r_S{}'$ which can pass the verification equation $ch_2{}' = r_S{}' \oplus H(g^{xy'}, IDPW_{dig}, r_C)$ of the client.

Second, we consider the scenario of impersonation of client. Suppose that an adversary tries to impersonate the client. The adversary would intercept y_1, ch_1, and ch_2 in step (2). The adversary then tries to recover $g^y = y_1 \oplus H(ID, IDPW_{dig})$ for getting g^{xy}. However, since the adversary does not know the hashed value $IDPW_{dig}$, he cannot recover the correct value of g^y. Therefore, the adversary cannot impersonate the client as well as the server.

5.3 Security against Denial-of-Service attack

We show that our protocol is secure against Denial-of-Service attack in password change protocol. Suppose that the adversary intercepts $\{ID, m\_IDPW_{dig}, mac\}$ in step (5). He can compute $IDPW_{dig}{}'$ with a new password PW_{new} chosen by the adversary. He then tries to make valid values

$$mask' = H(g^{xy}, r_C, r_S),$$

$$IDPW_{dig}{}' = mask \oplus m\_IDPW_{dig}, \text{ and}$$

$$mac' = H(g^{xy}, IDPW_{dig}{}', r_C, r_S).$$

However, while g^x can be intercepted in step (1), g^y cannot be recovered by anyone except the genuine client. Therefore, the adversary cannot compute the DH-value g^{xy}, and accordingly, he cannot make valid values mac', $mask'$, and $IDPW_{dig}{}'$.

Ⅵ. Conclusions

We discussed the security flaws of Hölbl *et al.*'s password-based user authentication and password change protocols based on Munilla and Peinado's paper. That is, Hölbl *et al.*'s remote user authentication scheme is vulnerable to off-line dictionary attacks and Denial-of-Service attack. Based on the observation of its insecure structure of the protocol, we provide a new secure remote authentication protocol and password change protocol which can resist impersonation attacks, off-line password guessing attack, and Denial-of-Service attack.

Up to now, many remote user authentication schemes are proposed without formal security proof. We are considering formal security proof of the proposed scheme for further research.

참 고 문 헌

〔1〕 S. Bellovin, and M.merritt, "Encrypted Key Exchange: Password-Based Protocols Secure against Dictionary Attacks," *Proc. of the Symposium on Security and Privacy*, IEEE Computer Society, pp. 72-84, 1992.

〔2〕 W. Diffie, and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, Vol. 22, No. 6, pp. 644-654, 1976.

〔3〕 M. Hölbl, T. Welzer, and B. Brumen, "Improvement of the Peyravian-Jefferies's user authentication protocol and password change protocol", *Computer Communications*, Vol. 31, No. 10, pp. 1945-1951, 2008.

〔4〕 J. Munilla, and A. Peinado, "Off-line password-guessing attack to Peyravian-Jeffries's remote user authentication protocol," *Computer Communications*, Vol. 30, No. 1, pp. 52-54, 2006.

〔5〕 J. Munilla, and A. Peinado, "Security flaw of Hölbl *et al.*'s protocol," *Computer Communications*, Vol. 32, No. 4, pp. 736-739, 2009.

〔6〕 M. Peyravian, and C. Jeffries, "Secure remote user access over insecure networks," *Computer Communications*, Vol. 29, No. 5, pp. 660-667, 2006.

〔7〕 K. A. Shim, "Security flaws of remote user access over insecure networks," *Computer Communications*, Vol. 30, No. 1, pp. 117-121, 2006.

이 지 선 (Ji-Seon Lee) 정회원

1991년 2월 서강대학교 전산학과

1998년 8월 서강대학교 컴퓨터공학과 석사

2008년 2월 서강대학교 컴퓨터공학과 박사

2008년 3월~현재 고려대학교 정보경영공학전문대학원, BK21 유비쿼터스 정보보호 사업단 연구교수

<관심분야> 암호학, 네트워크 보안, 콘텐츠 보안

박 지 혜 (Ji Hye Park) 정회원

2007년 2월 서강대학교 컴퓨터공학과졸업

2009년 2월 서강대학교 컴퓨터공학과석사

2009년 3월~현재 티맥스소프트 전임연구원

<관심분야> 암호학, 네트워크 보안

장 직 현 (Jik Hyun Chang) 정회원

1972년 2월 서울대학교 수학과 학사

1977년 8월 서울대학교 수학과 석사

1986년 8월 미네소타대학 전산학과박사

1986년 9월~현재 서강대학교 컴퓨터공학과 교수

<관심분야> 알고리즘 설계와 분석, 암호 알고리즘