ADVANCED SECURE USER AUTHENTICATION FRAMEWORK FOR CLOUD COMPUTING

Rui Jiang

School of Information Science and Engineering

Southeast University, Jiangsu 210096

Nanjing, ChinaS

Email: R.Jiang@seu.edu.cn

Submitted: Mar.17, 2013 ***Accepted: July 22, 2013*** ***Published: Sep.05, 2013***

Abstract—Cloud Computing, as an emerging, virtual, large-scale distributed computing model, has gained increasing attention these years. Meanwhile it also faces many security challenges, one of which is authentication. Lots of researches have been done in this area. Recently, Choudhury et al proposed a user authentication framework to ensure user legitimacy before entering into the cloud. They claimed their scheme could provide identity management, mutual authentication, session key agreement between the user and the cloud server, and demanded user password change. However, we find the scheme will easily suffer from some attacks such as the masquerading attack, the OOB (out of band) attack, and the password change flaw through our analysis. In this paper, we first point out the security vulnerabilities to the Choudhury et al's scheme, and present the detailed attacks on the scheme. Then, based on some remote user authentication schemes such as Ku-Chen's scheme and Chen's scheme, we apply the two-factor authentication technology to propose our advanced secure user authentication framework which can overcome above security shortages. Without sending one time key through secure OOB channel, our new protocol is able to ensure that only legitimate users can

access the cloud service based on smartcard. In addition, our advanced scheme can hold all the merits of the Choudhury et al's scheme. Formal security analysis, which is based on the strand space model and authentication test, proves that our proposed scheme is secure under standard cryptographic. Also, the simulation results illustrate that our advanced scheme is more efficient on the communication performance than other schemes.

Index terms*: Cloud Computing, Remote user authentication, Smartcard, Authentication, Formal method.*

I. INTRODUCTION

Recently, cloud computing has been greatly increased in both academic research and industry technology. It is like a "resource pool", which can provide the cost-effective and on-demand services to meet the needs by outsourcing data. In [1], cloud computing is defined as follows: "Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services."

The emerging of cloud computing allows companies to focus more on their core business and brings perceived economic and operational benefits [2]. However, as an attractive paradigm, it also faces many challenges, where the security issues are the most important. As mentioned in [3], cloud security issues can be classified into four categories: authentication, data integrity, data confidentiality and access control. User authentication is the paramount requirement for cloud computing that restricts illegal access of cloud server and so far many schemes have been proposed. Figure 1 shows the approximate frame of cloud computing. Usually, cloud computing system contains three parts: a data owner, a user and a cloud service provider. The data owner outsources the encrypted data to cloud and the authorized users request for the corresponding data. However, when the data owner or the user requests for the stored data in cloud, the authentication of the legality of the user identity is quite important. In this paper, we focus on the remote user authentication between the user and cloud server. The architecture of the remote user authentication is illustrated in figure 2.

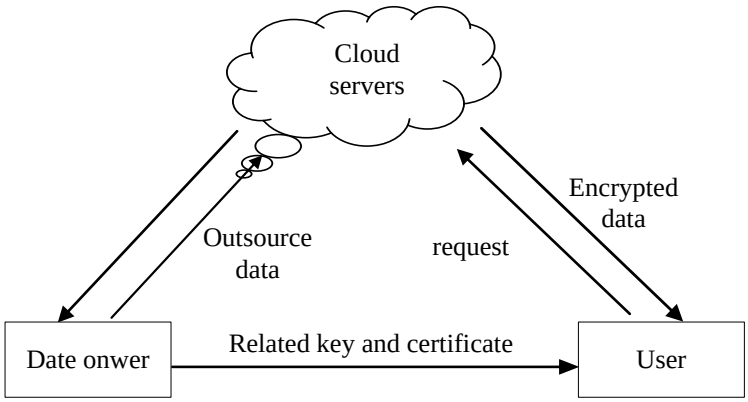


Figure 1. The structure of cloud access

In 1981, Lamport [5] proposed a remote user authentication system, in which, the server stores the hash value of the user's password for the later verification. However, in 2000, Hwang et al [6] pointed out that if the password table was compromised, the whole system could be invalid. Then they proposed a new remote user authentication scheme using smart cards. The scheme was based on the ElGamal's public cryptosystem and did not require a system to maintain a password table for verifying the legitimacy of the login user. But this scheme was not able to resist impersonate attack. A legitimate user could impersonate other valid user to use his ID and PW without knowing the secret key. In 2002, Chien et al [7] proposed an efficient password based remote user authentication scheme, and claimed that their scheme had the merits of providing mutual authentication, no verification table, freely choosing password, and involving only few hashing operations. In 2004, Ku–Chen [8] pointed out that Chien et al.'s scheme was vulnerable to a reflection attack [9], insider attack [10] and was not reparable. In 2010, Chen and Huang [11] proposed a user participation-based authentication combining CAPTCHA and visual secret sharing. Later

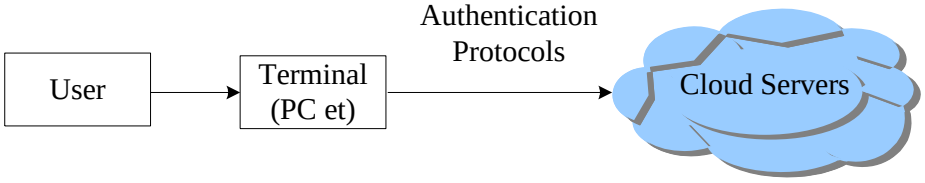


Figure 2. The architecture of remote user authentication

, Li et al[12] pointed out that Chen et al's scheme [11] would suffer from masquerading attack when the smartcard had been stolen. Recently, the user login security is more and more concerned in the case of smartcard lost [12][13].

In many circumstances, the weakest link is the password used to access a cloud-based application. The

reason is the password is often easy to guess or steal. To help combat any human-introduced weakness to the security equation, many security-focused services are deploying a technology called two-factor authentication. Rather than using just one password to login to a website, users couple a password with a second authentication mechanism. With two-factor authentication, even if someone has stolen your password, they'll need physical access to your secondary authentication mechanism in order to access your cloud-based data [4].

Choudhury et al [14] presented a user authentication frame for cloud computing. They proposed a new idea to apply remote user authentication with smartcard to cloud computing. They claimed their scheme verified user authenticity using two-step verification, which was based on password, smartcard and out of band authentication. However, through our security analysis, the scheme exists extremely serious attacks such as the masquerading attack, the OOB (out of band) attack, and the password change flaw.

In this paper, we focus on secure remote user authentication in cloud computing. We first analyze the vulnerability and attacks existing in Choudhury et al's protocol. To overcome these security shortages, based on some remote user authentication schemes such as Ku-Chen's scheme and Chen's scheme, we apply the two-factor authentication technology, which consists of the user password PW and the secret random number x, to propose an advanced secure authentication protocol which can provide mutual authentication, identity management, session key agreement between the user and the cloud server, and the demanded user password change without sending one time key through secure OOB channel. In our scheme, we actually realize the basic requirements for evaluating a password authentication scheme [15]. Based on the strand space model [18] and authentication test [19], we formally prove that our proposed scheme is secure under standard cryptographic. In addition, we make the performance simulation to illustrate that our advanced scheme is more efficient on the communication performance than other schemes.

The rest of the paper is organized as follows. Section 2 gives a brief introduction of the related authentication frame proposed by Choudhury et al and points out the vulnerabilities and attacks to the protocol. The new advanced secure user authentication scheme against smartcard security breach is proposed in Section 3 and formal proof and security analysis of our protocol are given in Section 4, respectively. Finally, we conduct a performance simulation in Section 5 and make a conclusion in Section 6.

II. REVIEW OF RELATED WOEKS

Recent years, a few password-based remote authentication schemes using smartcard have been proposed in cloud computing. In this section, we review one of the recent password-based remote authentication schemes, which is the Choudhury et al's scheme. For convenience, we list the common notations used throughout this paper as follows.

a. Notations

Notation	**Description**
A	A login user
S	The cloud server
ID	Identity of the user
PW	The password of the user
K	Onetime key
x	A user's secret number
y	A servers secret number stored at the server
p	A large prime number
g	Primitive element in the Galois field GF(p)
h(·)	One way hash function
$E_K(\cdot) / D_K(\cdot)$	The symmetric encryption/decryption function with key *K*
||	Concatenation operation
X→Y	Message M is sent X to Y through public channel
X=>Y	Message M is sent X to Y through secure channel
$\oplus$	The XOR operation

b. Brief review of Choudhury et al's Scheme

There are three phases and one activity in Choudhury et al's scheme [14], which are registration phase,

login phase, authentication phase and password change.

b.i Registration phase

The registration phase is consisted of two steps.

Step1: A=>S： ID, $h(PW \oplus x)$, $h(x)$

User A sends $\{ ID, h(PW \oplus x), h(x) \}$ to the cloud server through a secure channel, where x is a random number generated by A.

Upon receiving the message, cloud server check the ID firstly. If ID (new) = ID (existing), the server rejects the registration and goes back to the beginning. Otherwise, the server stores ID in the ID table, and computes $J = h(ID \oplus h(PW \oplus x))$, $I = h(ID \| y)$ and $B = g^{h(y)+h(I \| J)+h(x)} \bmod p$, where y is a random number generated by server. S stores $\{I, J, B, p, g, h(.)\}$ in the smartcard and sends smartcard to the user.

Step2: The cloud server sends the smartcard to the user A through a secure channel.

b.ii Login phase

Step1: User A inserts the smartcard and enters ID and PW.

Local system computes $J_1 = h(ID \oplus h(PW \oplus X))$, and check whether $J_1 = J$ or not. If true, it proceeds to the next step, otherwise it exits.

Step2: A→S: *B*, *C*

User A computes $C = h(I \| J)$ and sends $M_1 = < B, C >$ to the server.

Step3: S→A: $h(B^{"}), h(L)$

The server generates K and then computes $B^{"} = g^{C+h(y)} \bmod p$, $h(B^{"})$, $L = h(B^{"} \| K)$ and $h(L)$. S sends $M_2 = < h(B^{"}), h(L) >$ to the user using public channel. At the same time, S sends onetime key K to user's mobile phone using secure OOB channel.

Step4: A checks $h(B^{"}), h(L)$

Upon receiving M_2, A computes $B^{'} = Bg^{-h(x)} \bmod p$, $h(B^{'})$, $L^{*} = h(B^{'} \| K)$ and $h(L^{*})$. Then A checks whether $h(B^{'}) = h(B^{"}), h(L^{*}) = h(L)$ or not. If both conditions are true, then he proceeds to the next step. Otherwise, he terminates the login session.

b.iii Authentication phase

Step1: A→S: *I, h(R), T*

A computes $R = h(T \,\mathrm{P}\, B')$ and sends $M_3 =< I, h(R), T >$ to the server, where T is the timestamp of the current time.

Step2: S→A: $h(S_k)$

At First, the server checks if T' - $T \leq \Delta T$ holds true or not. If the condition is false, then the server rejects the session. Otherwise, it proceeds to the next step. Where ΔT is the maximum legal time difference for an authentication session defined for a networking system and T' is the current time stamp of the server. Then, S computes $I' = h(ID \,\mathrm{P}\, y)$ and $R^* = h(T \,\mathrm{P}\, B^*)$, and checks whether $h(R^*) = h(R)$ and $I' = I$. If both equations are true, S will generate $S_k = (R \oplus L)$ and sends the hash value of S_k to the user. If they are not true, the server terminates the communication.

At last, the user checks $h(S_k)$ by computing $h(S_k^*) = h(R \oplus L)$.

b.iv Password change

When the user A wants to change the password in the self system, he enters *ID* and *PW* , and computes $J^* = h(ID \oplus h(PW \oplus x))$. Local system will check $J^* = J$. if $J^* \neq J$, then it rejects the request, otherwise A enters a new password PW' and generates x'. The smartcard computes $J' = h(ID \oplus h(PW' \oplus x'))$ and replace J with J' and x with x' in the smartcard.

c. The Attacks

c.i Masquerading attack

When the user's smartcard is lost, stolen or got by an attacker, the attacker can extract the secret information stored in the smartcard. For the messages sent from A to S are only related with secret data stored in the smartcard, the attacker can masquerade as a legal user. The attacker can

compute $C = h(I \| J)$, $B^{'} = Bg^{-h(x)} \bmod p$, $h(R) = h(h(T \| B^{'}))$. Therefore, the messages in login phase step2 and authentication phase step1 can be generated by the attacker so that the attacker can successfully create a valid login request as a legal user.

In addition, at the end of the authentication phase, S sends $h(S_K)$ to A. If the attacker modifies $h(S_K)$, A can not have a chance to notify the server although he checks the modified $h(S_K)$ is wrong. Therefore, the consequence is that the server completes the authentication while the client doesn't think so. The communication between user and server cannot be established. Therefore, the mutual authentication fails.

c.ii OOB attack

In the scheme, the authors claimed the major advantage of the scheme was the OOB (out of band) factor. To improve the security, the cloud server generated the onetime key for the mobile network through HTTP/SMS gateway. The mobile network delivered the onetime key to the user's mobile phone via SMS. However, some facts show that this method is not as good as the authors hope.

Lots of attacks for out of band have been proposed such as SMS interception, phone flooding or SMS phishing. The details are described in [16], [17]. For example, in the login phase of the scheme, when the cloud server sends onetime key K to user's mobile phone by SMS channel (OOB), it is easy for the attacker to intercept the message. Therefore, it leads to disclose the onetime key. Also, in the cyber crime trend of future, these kinds of attacks will become a great threat for the out-of-band authentication.

Moreover, the scheme cannot ensure the real-time communication through SMS. Sometimes, due to network congestion and other problems, the messages cannot reach on time. Therefore, it's intolerable to the users.

c.iii Password change flaw

In the phase of password changing, the user only makes the change of J and x in the smartcard. However, the user does not change B , which is used in the authentication. This may lead to login failures once the user changes the password. For example, the original parameters were PW , J and x . After the user alters the password, these parameters are replaced with $PW^{'}$, $J^{'}$ and $x^{'}$. When the user logins into the cloud server, in the step3 of the login phase, the server computes

$B'' = g^{C+h(y)} \bmod p = g^{h(IPJ')+h(y)} \bmod p$. Then the server sends $h(B'')$ to the user for verification. The user computes $B' = Bg^{-h(x')} \bmod p = g^{h(IPJ)+h(y)+h(x)+h(x')} \bmod p$ in step4. Obviously, $h(B') \neq h(B'')$. The authentication fails and the user cannot login successfully.

III. OUR PROPOSED SCHEME

In this section, based on some remote user authentication schemes such as Ku-Chen's scheme and Chen's scheme, we apply the two-factor authentication technology, which consists of the user password PW and the secret random number x, to propose our advanced secure user authentication scheme. Our proposed scheme is able to resolve the security flaws of the Choudhury et al's scheme and hold all the merits of the scheme. Our new scheme has four phases which are the registration phase, the login phase, the authentication phase and the password change phase. The entire protocol process is shown in Figure 3. The details are described as follow.

a. Registration Phase

In the registration phase, user provides appropriate identification details to the cloud server. Then the cloud server issues a smartcard to the user according user's data.

1) User A selects a random number x and computes $h(PW \oplus x)$.

2) A=>S: User sends ID , $h(PW)$ and $h(PW \oplus x)$ to the cloud server through a secure channel.

3) S checks whether the ID has existed in server. If it is true, S rejects registration request. Otherwise, S generates y and computes $I = h(ID \,\mathrm{P}\, y), B = g^{ID+h(PW)+h(y)} \bmod p$.

4) S=>A: The cloud server sends a smartcard which contains $\{I, B, p, g, h(\cdot)\}$ to the user A through a secure channel.

5) A enters x into his smartcard. Now smartcard contains $\{I, B, p, g, h(\cdot), x\}$.

6) S stores ID and $h(PW \oplus x)$ in the server.

b. Login Phase

This phase is invoked when user wants to login into the cloud.

1) User A inserts his smartcard and enters ID and PW.

2) The smartcard computes $C = h(I \| h(PW \oplus x) \| T_u)$, where T_u denotes A's current timestamp.

3) A→S: ID, C, T_u.

c. Authentication Phase

After receiving the login request message $\{ID, C, T_u\}$, the server verifies the identity of the user. The procedure is as follows.

1）If $T_u^{'} - T_u < \Delta T$, S rejects A's login request. Otherwise, S performs the following computations $I^* = h(ID \| y)$, $C^* = h(I^* \| h(PW \oplus x) \| T_u)$, where $T_u^{'}$ is the current timestamp of server and ΔT is the

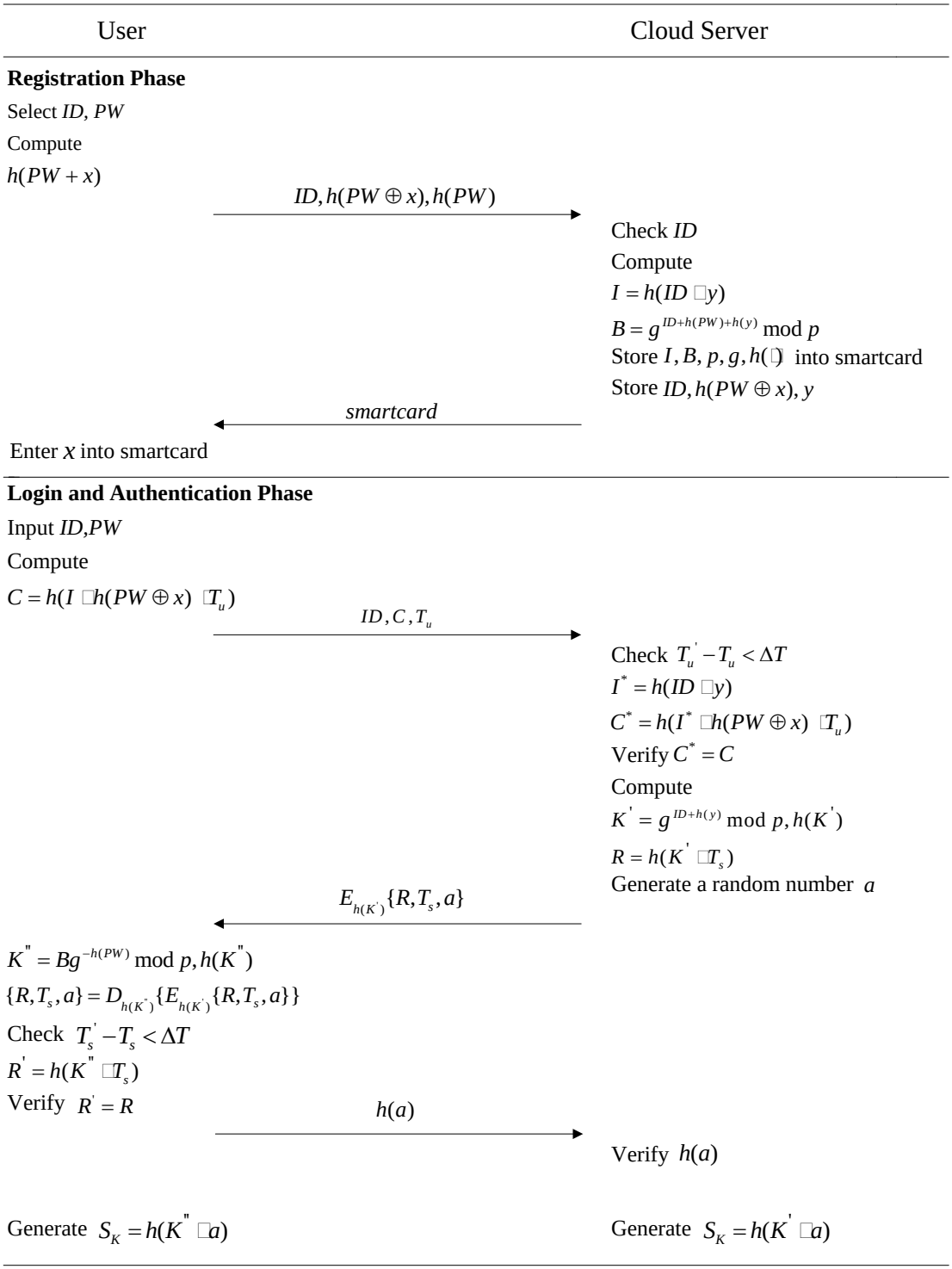


Figure 3. Our advanced secure user authentication scheme

maximum time interval for transmission delay. If C^* equals to C, S accepts A's login request and computes $K' = g^{ID+h(y)} \bmod p$, $h(K')$ and $R = h(K' \| T_s)$, where T_s is S's current timestamp. S generates a

random number a.

2) S→A: $E_{h(K')}\{R, T_s, a\}$.

3) A computes $K'' = Bg^{-h(PW)} \bmod p$ and $h(K'')$. Then A uses $h(K'')$ to decrypt $E_{h(K')}\{R, T_s, a\}$ and gets $\{R, T_s, a\}$. A checks the timestamp. If T_s is invalid, A terminates this session. Otherwise, A computes $R' = h(K'' \mathrm{P} T_s)$ and compares R' to the received R. If they are equal, A successfully authenticates S.

4) A→S: $h(a)$.

5) S checks $h(a)$. If $h(a)$ is correct, the mutual authentication successes. Now both user A and server S can compute the session key $S_K = h(K' \mathrm{P} a) = h(K'' \mathrm{P} a)$.

d. Password Change Phase

This phase is invoked when the user wants to change his password.

1) A insert his smartcard into smartcard reader and enter ID and PW.

2) A→S: $E_{S_K}\{h(PW \oplus x) \mathrm{P} h(PW' \oplus x) \mathrm{P} b\}$

A and S execute the login and authentication phase mentioned above. If A passes the verification, A will send a password change request to S, and then submit $h(PW \oplus x)$ and $h(PW' \oplus x)$, where PW' is A's new password, b is a random number.

3) S checks $h(PW \oplus x)$ and replaces it with $h(PW' \oplus x)$.

4) S→A: $h(b)$

5) A checks $h(b)$. If it is correct, the smartcard performs the following computations:

$$Z = Bg^{-ID-h(PW)} \bmod p, B' = Zg^{ID+h(PW')} \bmod p.$$

6) A replaces B with B' in the smartcard.

IV. SECURITY ANALYSIS

In this section, we formally analyze our advanced scheme based on strand space model [18] and authentication tests [19], and prove its confidentiality and authentication correctness.

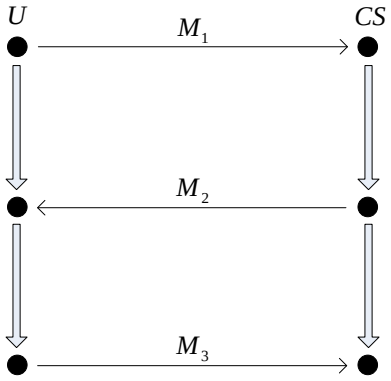
According to strand space model theory, at the login and authentication phase, our advanced secure user authentication protocol can be formalized as the following two types of regular strands (figure 4).

1）Initiator（U）strands with trace: $\left\langle +ID\{I\{PW\}_{K_x}T_u\}_{K'_{pub}}T_u, -\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}}T_s a\}_{K_{h(K')}}, +\{a\}_{K'_{pub}} \right\rangle$,

where $ID, I, PW, T_u, T_s, a \in \mathbf{T}$, $K_x, K_{pub}, K_{T_s}, K_{h(K')}, K'_{pub} \in \mathbf{K}$. $Init[ID, C, T_u, T_s, a]$ will denote the set of all strands with the trace shown.

2）Responder（CS）strands' trace in $\mathrm{Re}sp[ID, C, T_u, T_s, a]$ is: $\left\langle -ID\{I\{PW\}_{K_x}T_u\}_{K'_{pub}}T_u, +\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}}T_s a\}_{K_{h(K')}}, -\{a\}_{K'_{pub}} \right\rangle$

where $ID, I, PW, T_u, T_s, a \in \mathbf{T}$, $K_x, K_{pub}, K_{T_s}, K_{h(K')}, K'_{pub} \in \mathbf{K}$. $\mathrm{Re}sp[ID, C, T_u, T_s, a]$ will denote the set of all strands with the trace shown.



Where: $M_1 = ID\{I\{PW\}_{K_x}T_u\}_{K'_{pub}}T_u$

$M_2 = \{\{\{ID\}_{K_{pub}}\}_{K_{T_s}}T_s a\}_{K_{h(K')}}$

$M_3 = \{a\}_{K'_{pub}}$

Figure 4 . Normal bundle of our advanced authentication protocol

a. Protocol confidentiality

Proposition 1 Assume $\sum$ is the protocol's strand space, C is a bundle in $\sum$. C is consisted of s_U and s_{CS} . $s_U \in U[ID, I, PW, T_u, T_s, a]$ with C -height 3. $s_{CS} \in CS[ID, I, PW, T_u, T_s, a]$ with C -height 3. $K'^{-1}_{pub} \notin K_P$,

and PW is unique originating in Σ. Let $S = \{K'^{-1}_{pub}, PW\}$, $\underline{K}$ = (**K**/ S)$^{-1}$, for any normal node $n \in C$, $term(n) \notin I_{\underline{K}}[S]$.

Proof. We apply the proof by contradiction. Suppose that there exists a normal node $n \in C$. Let $term(n) \in I_{\underline{K}}[S]$, then at least one of K'^{-1}_{pub} and PW is item of $term(n)$. According to the definition, neither U nor CS contains the item K'^{-1}_{pub}. So PW must be the item of $term(n)$.

If n is positive sign, $ObjectID \sqsubset term(n)$ means $n = \langle s_U, 1 \rangle$. As PW is unique generated in Σ, $\{I\{PW\}_{K_x} T_u\}_{K'_{pub}} \in I_{\underline{K}}[S]$. This is $K'_{pub} \in \underline{K}$. This conclusion is in contradiction with the prerequisite $\underline{K}$ =（**K** / S ）$^{-1}$. Therefore the proposition is proved. ■

Proposition 1 proves the confidentiality of PW depending on the assumption that $K'^{-1}_{pub} \notin K_P$. In the same way, we can prove the confidentiality of y. Therefore, the secret data between the user and the cloud server can gain effective protection. Proposition 1 realizes secure key agreement between the user and the cloud server, which is the premise of the mutual authentication.

b. Authentication correctness

Proposition 2 Let C be a bundle in Σ, and s be an initiator's strand in $Init[ID, I, PW, T_u, T_s, a]$ with C-height 2. Assume $K_{T_s} \notin K_p$, and suppose that ID is uniquely originating. Then there must be a responder strand $s' \in \mathrm{Re}sp[ID, C, T_u, T_s, a]$ with C-height 2.

Proof. We show first that the first and second nodes of s construct an incoming test for ID. $\{\{ID\}_{K_{pub}}\}_{K_{T_s}}$ is a test component for ID in $\langle s, 2 \rangle$, because it contains ID, and no regular node has any term of this form as a proper subterm. Checking the assumptions, it follows that $\langle s, 1 \rangle \Rightarrow^{+} \langle s, 2 \rangle$ is an incoming test for ID in $\{\{ID\}_{K_{pub}}\}_{K_{T_s}}$.

By Incoming Test, there exist regular nodes $n_0, n_1 \in C$ such that $\{\{ID\}_{K_{pub}}\}_{K_{T_s}}$ is a component of n_1 and $n_0 \Rightarrow^{+} n_1$ is a transforming edge for ID.

Because n_1 is a positive regular node and $\{\{ID\}_{K_{pub}}\}_{K_{T_s}} = term(n_1)$, ID is uniquely originating in $\langle s, 1 \rangle$, then there must exist a negative regular node n_0 to receive ID. Since n_0 is a negative node, n_0 is

$\langle s',1\rangle$ for some responder strand $s' \in \mathrm{Re}\,sp[ID,C,T_u,T_s,a]$. Since $\langle s',1\rangle \Rightarrow^{+} \langle s',2\rangle$ and $term(\langle s',2\rangle) = \{\{ID\}_{K_{pub}}\}_{K_{T_s}}$, we see that $T_u' = T_u$ and $T_s' = T_s$. Therefore the C-height of s' is 2. ■

Proposition 2 proves the security certificate of CS depending on the assumption that $K_{T_s} \notin K_p$. In addition, since our advanced protocol contains an incoming test for ID, it guarantees the recency of ID. Thus our advanced protocol can prevent any malicious active and passive attack in the first two steps.

Proposition 3 Let C be a bundle in Σ, and s be a responder's strand in $\mathrm{Re}\,sp[ID,C,T_u,T_s,a]$ with C-height 3. Assume $K_{h(K')} \notin K_p$, and suppose that a is uniquely originating. Then there must be an initiator strand $s' \in Init[ID,C,T_u,T_s,a]$ with C-height 3.

Proof. We show first that the second and third nodes of s construct an outgoing test for a. $\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}} T_s a\}_{K_{h(K')}}$ is a test component for a in $\langle s,2\rangle$, because it contains a, and no regular node has any term of this form as a proper subterm. Checking the assumptions, it follows that $\langle s,2\rangle \Rightarrow^{+} \langle s,3\rangle$ is an outgoing test for a in $\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}} T_s a\}_{K_{h(K')}}$.

By Outgoing Test, there exist regular nodes $n_0, n_1 \in C$ such that $\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}} T_s a\}_{K_{h(K')}}$ is a component of n_0 and $n_0 \Rightarrow^{+} n_1$ is a transforming edge for a.

Because n_1 is a positive regular node and $\{\{\{ID\}_{K_{pub}}\}_{K_{T_s}} T_s a\}_{K_{h(K')}} = term(n_1)$, a is uniquely originating in $\langle s,2\rangle$, then there must exist a negative regular node n_0 to receive a. Since n_0 is a negative node, n_0 is $\langle s',2\rangle$ for some initiators strand $s' \in Init[ID,C,T_u,T_s,a]$. Since $\langle s',2\rangle \Rightarrow^{+} \langle s',3\rangle$ and $term(\langle s',3\rangle) = \{\{\{ID\}_{K_{pub}}\}_{K_{T_s}} T_s a\}_{K_{h(K')}}$, we see that $T_s' = T_s$. Therefore the C-height of s' is 3. ■

Proposition 3 proves the security certificate of U depending on the assumption that $K_{h(K')} \notin K_p$. In addition, since our advanced protocol contains an outgoing test for a, it guarantees the recency of a. Thus our advanced protocol can prevent any malicious active and passive attack in the last two steps.

Combining the proof of proposition 2 and 3, we achieve the secure mutual authentication between users and cloud server. Our scheme can resist relay attack, man in the middle attack and so on. For example, the transmitted messages $C = h(I \,\mathrm{P}\, h(PW \oplus x) \,\mathrm{P}\, T_u)$, $R = h(K' \,\mathrm{P}\, T_s)$ contain timestamp, hence our scheme is strong against replay attack. If ID is modified as ID^* in login phase, then $I^* = h(ID^* \,\mathrm{P}\, y)$, $C^* = h(I^* \,\mathrm{P}\, h(PW \oplus x) \,\mathrm{P}\, T_u)$. Hence $C^* \neq C$, the communication will terminate. Moreover, no

matter which message is modified by an adversary, the communication will terminate. Hence man in the middle attack is resisted.

c. Security advantages

Compared to Choudhury et al's scheme, our advanced protocol has greatly enhanced the security in the following aspects.

Withstanding masquerade attack: our proposed scheme can withstand masquerade attack with smartcard information disclosing. When user A's smartcard has been stolen, the attacker can breach the data $I, B, p, g, h(\cdot)$ stored in the smartcard. However, the attacker cannot compute correct $h(I \| h(PW \oplus x) \| T_u)$ according to these parameters. Also, neither K' nor K'' can be got by the attacker without knowing PW or y. So even if the smartcard is stolen, our protocol can protect users' login security.

In addition, our proposed scheme can provide mutual authentication. At the step 2 of authentication phase, S sends $E_{h(K')}\{R, T_s, a\}$ to A. A checks R to verify S. Meanwhile A sends a response $h(a)$ to S for verification. Thus the mutual authentication is performed.

Avoiding OOB attack: our proposed scheme does not use onetime key K. Instead, we use $h(K')$ to encrypt the message to ensure protocol secure. Thus we avoid transmitting K through OOB channel and avoid OOB attack. In Choudhury et al's scheme, the final session key depends on the onetime key K, which leads to the different final session key in every login. Although we don't use onetime key K, our final session key is based on a random number, which is different in every session. Therefore, in our advanced scheme, a different session key will be generated between user and server in every login.

Password change: our proposed scheme facilitates users to change password. As described in password change flaw, Choudhury et al's scheme cannot achieve the real function of password changing. In our scheme, when we change the password, we change both $h(PW \oplus x)$ in the server and B in the smartcard at the same time for the later authentication. It is inherently stronger compared static password based scheme.

For clearly seeing the advantages of security for our proposed scheme, we list a table compared with other schemes. The √ in the blank means corresponding scheme can withstand the related attack or meet the description.

Table I: Security performance comparison among related schemes

	Our scheme	Choudhury's[14]	Chien's[7]	Ku–Chen's[8]	Chen's [11]
Mutual authentication	√		√		√
Insider attack	√	√		√	√
Man-in-the-middle attack	√	√	√	√	√
Masquerade attack(stolen card)	√				
OOB attack	√		√	√	√
Insecure for changing password	√				√

V. PERFORMANCE SIMULATION

We make the performance simulation on our advanced protocol and other schemes with NS2. We assume the bandwidth of the wireless network is 2Mbps. In the simulation, we adopt AES as our encryption algorithm. The two metrics are the time delay and communication overhead which can show the performance characters when the protocol is running.

a. Communication performance for our scheme vs Choudhury's scheme

The time delay and communication overhead in the login and authentication phases for our protocol vs Choudhury's scheme are shown in figure 5 and figure 6. Figure 5 illustrates the total time delay when the protocols run 500 times, 1000times, 2000times, 3500times and 5000 times, respectively. The corresponding time delays of our advanced scheme vs Choudhury's scheme are 191.15s vs 334.50s, 388.24s vs 671.99s, 780.59s vs 1347.33s, 1367.04s vs 2351.47s, 1954.49s vs 3371.96s, respectively. Obviously, the time delay for our scheme is shorter than that for Choudhury's scheme. Our advanced

scheme is more efficient than Choudhury's scheme. The main reason is our scheme only has three messages during the login and authentication phases and Choudhury's scheme has four messages. Also, we do not count the time delay for Choudhury's scheme to send onetime key through OOB channel. If the onetime key sending process is included, the time delay for our scheme is much shorter than Choudhury's scheme.

Figure 6 shows the total communication overhead during the login and authentication phases when the protocols run 500 times, 1000 times, 2000times, 3500times and 5000 times, respectively. The corresponding communication overhead for our advanced scheme vs Choudhury's scheme is 12.56Mb vs 25.48Mb, 26.71Mb vs 50.89Mb, 54.13Mb vs 103.21Mb, 90.77Mb vs 180.33Mb, 130.49Mb vs 256.27Mb, respectively. Also, the average communication overhead for our advanced scheme is much less than that for Choudhury's scheme. The reason is the decrease of the interactive messages and computation for our advanced scheme. Hence the more the users login into the system, the more efficient our advanced scheme will be. Overall, our protocol can not only improve the security performance but also make the communication performance better.

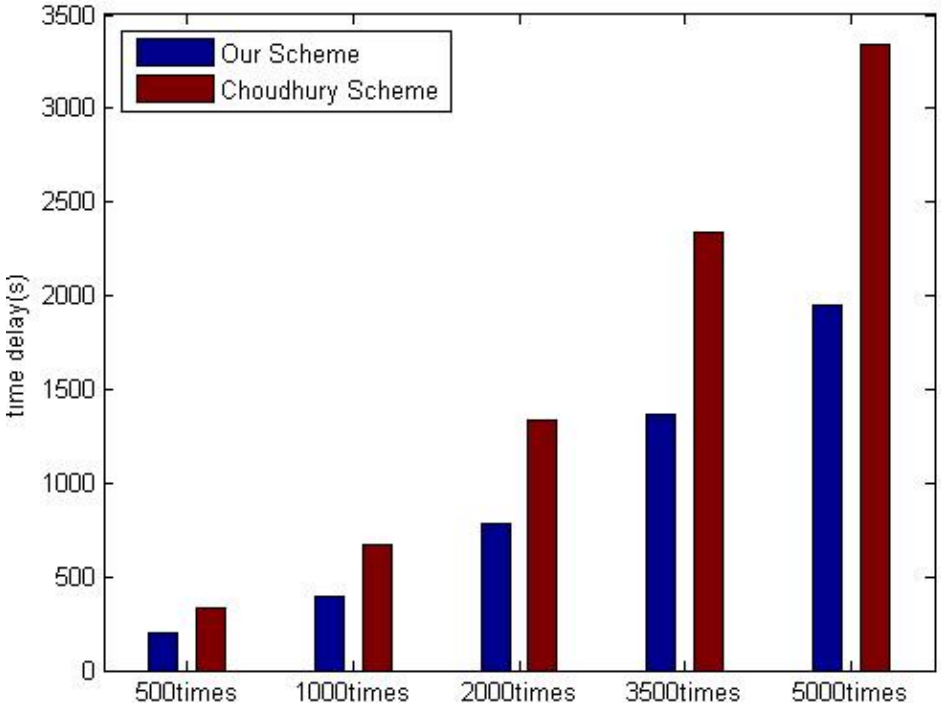


Figure 5. Time delay for our scheme vs Choudhury's scheme

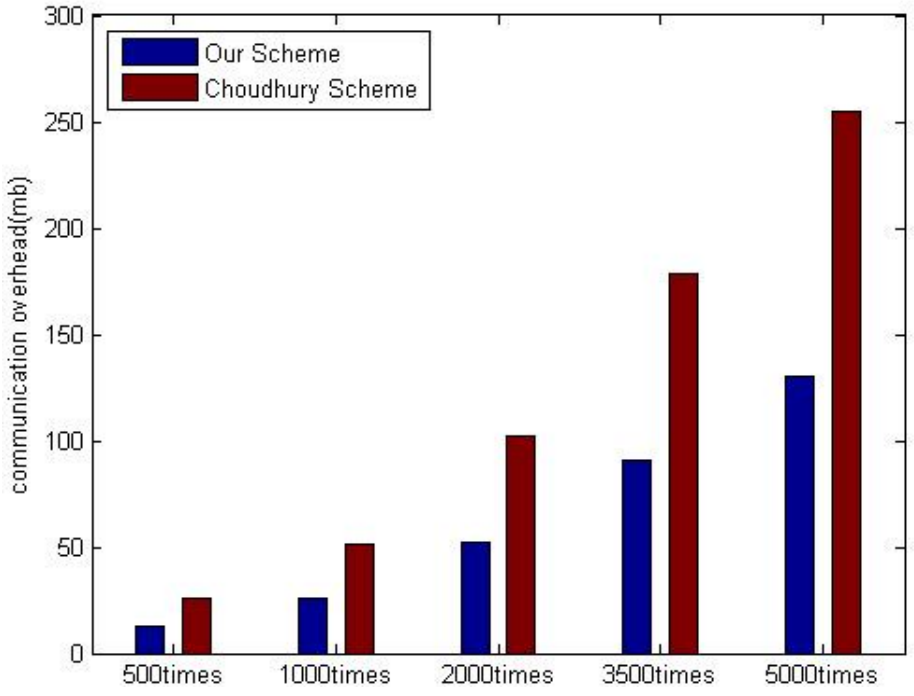


Figure 6. Communication overhead for our scheme vs Choudhury's scheme

b. Communication performance for our scheme vs Ku-Chen's scheme

The time delay and communication overhead of our protocol vs Ku-Chen's protocol are shown in figure 7 and figure 8, respectively. Figure 7 illustrates the total time delay when the protocols run 500 times, 1000times, 2000times, 3500times and 5000 times, respectively. The corresponding time delays of our advanced scheme vs Ku-Chen's scheme are 191.15s vs 130.10s, 388.24s vs 261.18s, 780.59s vs 522.24s, 1367.04s vs 912.48s, 1954.49s vs 1303.78s, respectively. The time delay of Ku-Chen's scheme is smaller than that of ours. The first reason is the Ku-Chen's scheme only has two interactive messages in authentication process, and the second reason is it only applies some simple algorithms such as XOR and hash. However, with the two messages for authentication, Ku-Chen's scheme can not ensure the mutual authentication between the user and the server. It can only guarantee the user authenticates the server. Therefore, it easily suffers from the masquerade attack and has the security shortage at the password change phase.

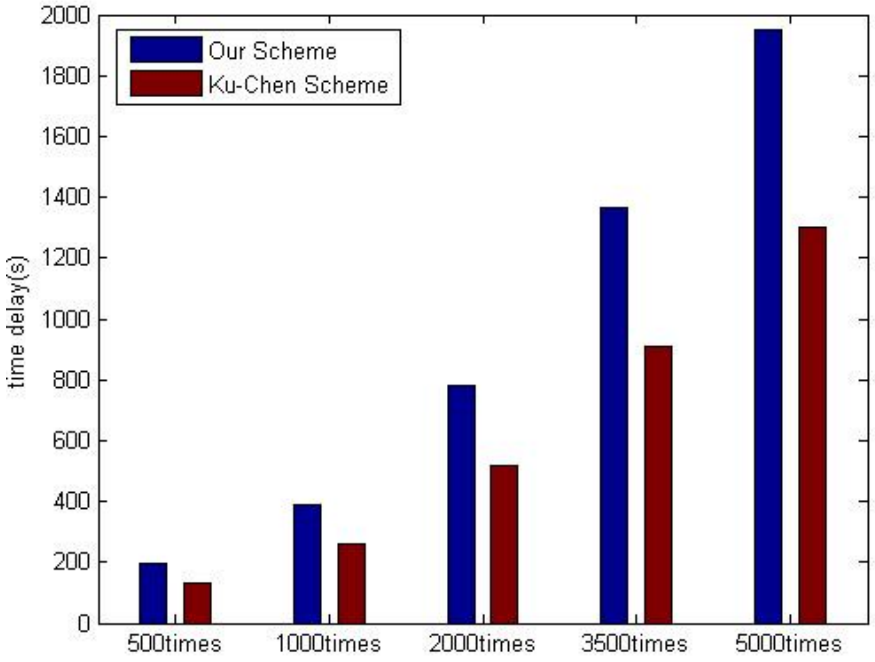


Figure 7. Time delay for our scheme vs Ku-Chen's scheme

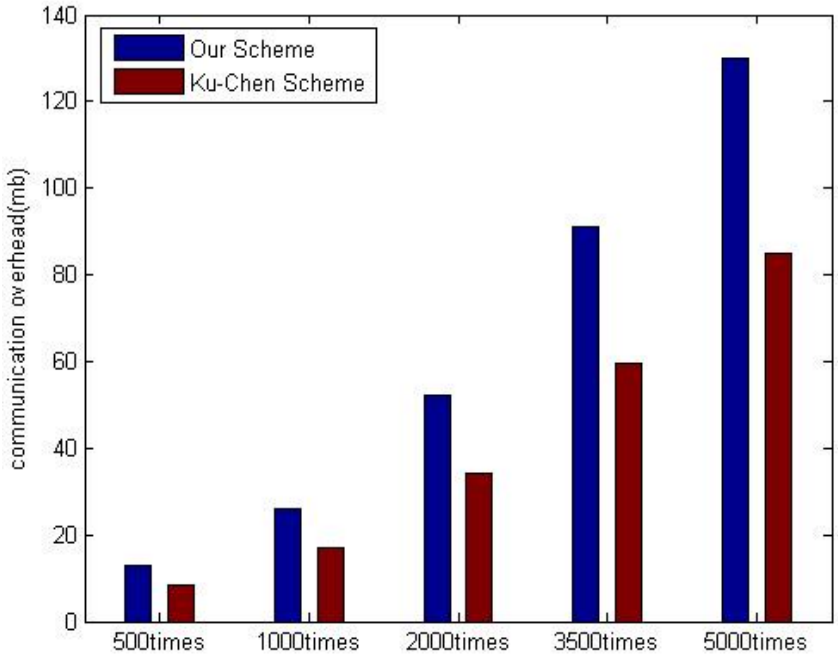


Figure 8. Communication overhead for our scheme vs Ku-Chen's scheme

Figure 8 shows the communication overhead between our scheme and Ku-Chen's scheme when the protocols run 500 times, 1000times, 2000times, 3500times and 5000 times, respectively. The simulation

results disclose the communication overhead for Ku-Chen's scheme is 8.50Mb, 16.97Mb, 34.45Mb, 59.99Mb, 86.17Mb, respectively. Comparing to our scheme, the Ku-Chen's scheme also seems more efficient. The two reasons are described above. The less messages and simple encryption for the Ku-Chen's scheme, on the one hand, may make the scheme more efficient. However, On the other hand, it will degrade the security level of the scheme.

c. Communication performance for our scheme vs Chen's scheme

With the same two sets of experiments above, figure 9 and figure 10 illustrate the time delay and the communication overhead of our advanced scheme vs Chen's scheme. The time delay for Chen's scheme is 4488.42s, 8980.83s, 17992.62s, 31473.91s, 45337.16s when the protocol runs 500 times, 1000times, 2000times, 3500times and 5000 times, respectively. While the communication overhead for Chen's scheme is 6660.51Mb, 13410.52Mb, 26712.32Mb, 46712.54Mb, 66675.39Mb when the protocol runs 500 times, 1000times, 2000times, 3500times and 5000 times, respectively. Obviously, Both the time delay and communication overhead for Chen's scheme are far larger than that for our advanced scheme. The main reason is that Chen's scheme combines CAPTCHA and visual secret sharing during the authentication process which causes users and servers to achieve verification by transmitting large encrypted image data. Therefore the communication overhead is very huge.

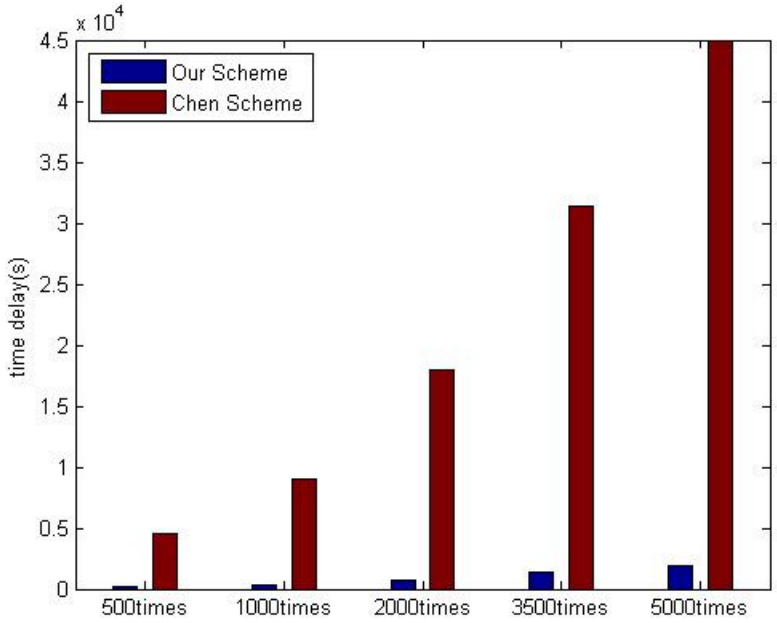


Figure 9. Time delay for our scheme vs Chen's scheme

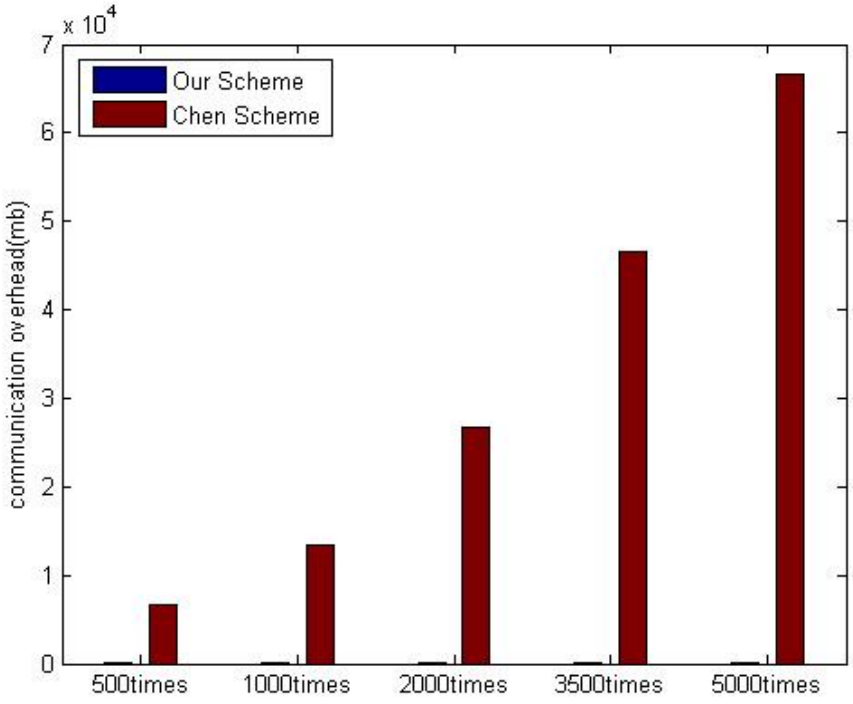


Figure 10. Communication overhead for our scheme vs Chen's scheme

d. Computation performance for our scheme vs other schemes

In addition to communication performance, the computation of the scheme is also an important factor. Table II lists the number of hash calculation, index calculation and encrypt/decrypt calculation during the login and authentication phases. Our advanced scheme has 10 times of hash calculation, 2 times of index calculation, 2 times of symmetric encryption, and 1 time of XOR calculation comparing to the 15 times of hash calculation, 2 times of index calculation, and 3 time of XOR calculation for the Choudhury's scheme, the 7 times of hash calculation and 4 time of XOR calculation for the Ku-Chen's scheme, and the 5 times of hash calculation, 2 times of symmetric encryption, and 6 time of XOR calculation for the Choudhury's scheme. The overall computation for the three schemes is at the same level except the Ku-Chen's scheme. Therefore, our advanced scheme makes little influence on computation overhead.

Table II : The number of operations of the various algorithms

	Hash	Index calculation	Symmetric encryption	XOR
Our advanced scheme	10	2	2	1
Choudhury's scheme	15	2	0	3
Ku-Chen's scheme	7	0	0	4
Chen's scheme	5	0	2	6

VI. CONCLUSION

In this paper, we first analyze the vulnerability and attacks existing in Choudhury et al's protocol. To overcome these security shortages, based on some remote user authentication schemes such as Ku-Chen's scheme and Chen's scheme, we apply the two-factor authentication technology, which consists of the user password PW and the secret random number x, to propose an advanced secure authentication protocol which can provide mutual authentication, identity management, session key agreement between the user and the cloud server, and the demanded user password change without sending one time key through secure OOB channel. Our advanced scheme can hold all the merits of Choudhury's scheme and enhance the security for user communicating with cloud server. Then we formally prove that our proposed scheme is secure under standard cryptographic based on the strand space model and authentication test. Finally, we make the performance simulation to illustrate that our advanced scheme is more efficient on the communication performance than other schemes.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (NO.61202448), the Key Laboratory Hi-Tech Program of Changzhou City (No.CM20103003), the Key Laboratory Program of Information Network Security of Ministry of Public Security (No.C12602), and the Science and Technology Supporting Project of Changzhou City (No.CE20120030).

REFERENCES

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia, "A view of cloud computing", *Communications of the ACM*, 53 (4), 2010, pp.50-58.

[2] Hassan Takabi, James B. D. Joshi, Gail-Joon Ahn, "Secure Cloud: Towards a Comprehensive Security Framework for Cloud Computing Environments", *Proceedings of 34th IEEE Conference Workshops on Computer Software and Applications*, 19-23 July, 2010, Pittsburgh, PA, USA, pp. 393-398.

[3] Chun-Ting Huang, Zhongyuan Qin, C.-C. Jay Ku, "Multimedia Storage Security in Cloud computing: An Overview", *Proceedings of 13th IEEE International Workshop on Multimedia Signal Processing*, 17-19 October, 2011, Los Angles, CA, USA, pp.1-6.

[4]Jack Newton, "Beyond Passwords: Two Factor Authentication Comes to the Cloud", http:// www.slaw.ca /2010 /09/20/, 2010.

[5] L. Lamport, "Password authentication with insecure communication", *Communications of the ACM*, 24 (11), 1981, pp. 770–771.

[6] M.S. Hwang, and L.H. Li, "A New Remote User Authentication Scheme using Smart Cards", *IEEE Transactions on Consumer Electronics*, 46 (1), 2000, pp.28-30.

[7] H.Y. Chien, J.K. Jan, Y.M. Tseng, "An efficient and practical solution to remote authentication smart card", *Computers & Security*, 21(4), 2002, pp. 372–375.

[8] W.C. Ku, S.M. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards", *IEEE Transactions on Consumer Electronics*, 50 (1), 2004, pp.204–207.

[9] C. Mitchell, "Limitations of challenge-response entity authentication", *Electronic Letters*, 25 (17), 1989, pp. 1195–1196.

[10] W.C. Ku, C.M. Chen, H.L. Lee, "Cryptanalysis of a variant of Peyravian–Zunic's password authentication scheme", *IEICE Transactions on Communication*, E86-B (5), 2003, pp.1682–1684.

[11] T. H. Chen and J. C. Huang, "A novel user-participating authentication scheme," *The Journal of Systems and Software*, 83(5), 2010, pp.861–867.

[12] Chun-Ta Li, Cheng-Chi Lee, "A robust remote user authentication scheme using smart card", *Information Technology and Control*, 40 (3), 2011, pp. 236-245.

[13] H. C. Hsiang and W. K. Shih, "Weaknesses and improvements of the Yoon-Ryu-Yoo remote user

authentication scheme using smart cards," *Computer Communications*, 32(4), 2009, pp. 649–652.

[14] Amlan Jyoti Choudhury, Pardeep Kumar, Managal Sain, Hyotaek Lim, Hoon Jae-Lee, "A Strong User Authentication Framework for Cloud Computing", *Proceedings of 2011 IEEE Asia-Pacific Services Computing Conference*, Jeju, South Korea, December 12-15, 2011, pp.110-115.

[15] I-En Liao, Cheng-Chi Lee, Min-Shiang Hwang, "A password authentication scheme over insecure networks", *Journal of Computer and System Sciences* , 72(4), 2006, pp.727-740.

[16] S21sec, "ZeuS Mitmo: Man-in-the-mobile", http: //securityblog.s21sec.com/2010/09/zeus–man – in – mobile –i.html, 2011.

[17] Szu-yu Lin, "Enhancing the security of out-of-band one-time password two factor authentication in cloud computing", http://pc01.lib.ntust.edu.tw/ETD-db/ETD-search/view_etd?URN=etd-0720111-153542, 2011.

[18] F. T. Fábrega, J. Herzog, and J. Guttman, "Strand spaces: Proving security protocols correct," *Journal of Computer Security*, 7(2/3), 1999, pp.191–230.

[19] J. D. Guttman and F. J. Thayer Fabrega, "Authentication tests and the structure of bundles", *Theoretical Computer Science*, 283(2), 2002, pp.333-380.