Engineering, 2014, 6, 261-267
Published Online May 2014 in SciRes. http://www.scirp.org/journal/eng
http://dx.doi.org/10.4236/eng.2014.66030

Scientific Research

An Enhanced Remote User Authentication Scheme

Xiaohui Yang, Xinchun Cui, Zhenliang Cao, Ziqiang Hu

College of Information Technology and Communication, Qufu Normal University, Rizhao, China
Email: yy18769355005@gmail.com

Received 2 March 2014; revised 2 April 2014; accepted 9 April 2014

Abstract

Remote user authentication schemes are used to verify the legitimacy of remote users' login request. Recently, several dynamic user authentication schemes have been proposed. It can be seen that, these schemes have weaknesses because of using timestamps. The implement of strict and safe time synchronization is very difficult and increases network overhead. In this paper, we propose a new dynamic user authentication based on nonce. Mutual authentication is performed using a challenge-response handshake between user and server, and it avoids the problems of synchronism between smart card and the remote server. Besides, the scheme provides user's anonymity and session key agreement. Finally, the security analysis and performance evaluation show that the scheme can resist several attacks, and our proposal is feasible in terms of computation cost and communication cost.

Keywords

Password, Dynamic ID, User Authentication, Smart Cards, Nonce

1. Introduction

With the large-scale proliferation of internet and network technologies, people are able to access any service from any place and at any time. Remote user authentication schemes are used to verify the legitimacy of remote user's login request. Password-based authentication scheme is one of the convenient and efficient authentication mechanics. However, password-based authentication scheme suffers from attacks due to the low entropy password, thus designing a more secure and efficient authentication protocol is in urgent need. In 1981, Lamport proposed a remote user authentication scheme with password table [1]. Afterwards, several schemes and improvements [2]-[4] have been extensively proposed. However, most of them using the static identity (ID) are included. Since the user's login ID is static in these verifier-free schemes, it may leak partial information about the user's login messages so that the adversary can use it to forge the user's login messages by some subtle means.

How to cite this paper: Yang, X.H., *et al.* (2014) An Enhanced Remote User Authentication Scheme. *Engineering*, **6**, 261-267. http://dx.doi.org/10.4236/eng.2014.66030

One of the solutions to the problem is to employ dynamic ID in different login.

In 2004, Das *et al*. [5] proposed a dynamic ID-based remote user authentication scheme, which can resist replay, masquerade, and insider attacks. However, Wang *et al*. in 2009 [6] pointed out that Das *et al*.'s scheme is susceptible to smart card attack and does not provide mutual authentication. Then, Wang *et al*. proposed a more efficient and secure dynamic ID-based remote user authentication scheme. Recently, Khan *et al*. in 2011 [7] pointed out that Wang *et al*.'s scheme has insider attack and does not provide user's anonymity and session key agreement. Then, they proposed a dynamic ID based remote user authentication scheme. We can see that these schemes have weaknesses because of using timestamps and lead to serious clock synchronization problems. In this paper, we proposed an enhanced dynamic ID-based remote user authentication scheme. In this scheme, mutual authentication is performed using a challenge-response handshake between user and server, and it avoids the problems of synchronism. Furthermore, the scheme provides user's anonymity and session key agreement.

The remainder of this paper is organized as follows. In Section 2, we present an enhanced remote user authentication scheme. In Section 3, there is the analysis about this scheme. Finally, conclusions are presented in Section 4.

2. The Proposed Scheme

Although the implement of strict and safe time synchronization is very difficult and increases network overhead, most time synchronization schemes were not designed with security in mind. In addition, if the setting of the interval of transmission delay is too short, it will cause the failure of the legal users' login. However, if the setting of the interval of transmission delay is too large, it will be suffered from the relay attacks. Therefore, authentication protocols based on the timestamps not only introduces more safety risk, but also is unpractical. In this section, we propose an enhanced remote user authentication scheme. To avoid the clock synchronization problem, we replace the timestamp design with a novel nonce-based mechanism in our scheme. The improved scheme is divided into four phase: registration phase, login phase, authentication phase, and password change phase. Detailed steps of these phases of the proposed scheme are described as follows. The notations used throughout this paper are in **Table 1**.

2.1. Registration Phase

A user $\mathrm{U_i}$ with identifier ID_i should first carry out this phase once before he can use any of the services provided by the server S. In this phase, $\mathrm{U_i}$ and S need to perform the following steps.

Step R1. User $\mathrm{U_i}$ keys his identity ID_i and password PW_i, and his smart card computes and submits $\{ID_i, h(ID_i \| PW_i)\}$ to S, through a secure channel.

Step R2. After receiving the request, S computes $A_i = h(h(ID_i) \oplus x)$, $B_i = h(ID_i \| PW_i) \oplus A_i$ and $C_i = h(A_i)$, where x is the permanent secret key of S. Then, S sends $\{h(\cdot), B_i, C_i\}$ to $\mathrm{U_i}$ through a secure channel.

2.2. Login Phase

Whenever $\mathrm{U_i}$ wants to login a server S, he must perform the following steps:

Step L1. After inserting his smart card into the card reader, $\mathrm{U_i}$ inputs the identity ID_i and password PW_i. Then, the smart card computes $D_i = B_i \oplus h(ID_i \| PW_i)$, and $E_i = h(D_i)$.

Table 1. Notations.

Symbol	Description
$\mathrm{U_i}$	User i
S	Server
ID_i	Identity of the user i
PW_i	Password of the user i
$h(.)$	A secure hash function
x	Secret value of server
y	Secret value of server
$\oplus$	Bitwise XOR operation
$\|$	Concatenation operation

Step L2. The smart card checks whether or not E_i and C_i are equal. If yes, U_i passes the legitimate verification, and performs the following steps; otherwise, U_i is rejected.

Step L3. The smart card randomly chooses a nonce R_1 and computes $F_i = D_i \oplus R_1$.

Step L4. U_i sends the login request message $\{h(ID_i), F_i\}$ to the remote server S.

2.3. Authentication Phase

A user performs the remote authentication phase based on the login message for authentication as long as it visits the server. U_i and S perform the following steps to achieve mutual authentication and to establish a session key.

Step A1. After receiving the login message $\{h(ID_i), F_i\}$, S computes $G_i = h(h(ID_i) \oplus x)$ and $R_1' = F_i \oplus G_i$. Then, S chooses a nonce R_2 and computes $H_i = G_i \oplus R_2$.

Step A2. The server S sends the mutual authentication message $\{H_i, h(R_1')\}$ to the user U_i.

Step A3. After receiving the mutual authentication message $\{H_i, h(R_1')\}$ from the server S, the user U_i checks whether or not $h(R_1')$ and $h(R_1)$ are equal . If no, U_i rejects this message and terminates the operation; otherwise, U_i authenticates S successfully and computes $R_2' = H_i \oplus D_i$. Then, U_i sends $\{h(R_2')\}$ to S.

Step A4. When the server S receives $h(R_2')$, checks whether or not $h(R_2')$ and $h(R_2)$ are equal. If no, S sends reject message to the U_i; otherwise, S authenticates U_i.

After finishing mutual authentication phase, the user U_i and the server S each can compute a common session key $SK = h(R_1 \| R_2)$ for the next data transmission.

2.4. Password Change Phase

The user U_i can change his password without the help of the server S, and the details of the password change procedures are as follows:

U_i inserts the smart card, and input his old password pw_i and the identity ID_i. Then, the smart card computes $A_i' = B_i \oplus h(ID_i \| PW_i)$, $C_i' = h(A_i')$, and checks whether or not C_i' and C_i are equal. If the verification process is correct, the smart card asks the cardholder to resubmit a new password PW_i^{new}, and then smart card computes $B_i^{new} = h(ID_i \| PW_i^{new}) \oplus A_i$. At last, the smart card replaces the values of B_i stored in its memory with B_i^{new} to finish the password change phase.

3. Security Analysis

In this subsection, we present these security analyses of our scheme and show that proposed scheme can resist many kinds of attack. To analyze the security of our scheme, we assume that an attacker can obtain the secret values stored in the smart card by monitoring the power consumption [8] [9] and intercept the messages communicating between the user and the server.

3.1. User Anonymity

The proposed scheme can protect user's anonymity. In login phase, the user U_i will send the login request message $\{h(ID_i), F_i\}$ to the server S. Thus, the attacker might incept and analyze the login message. It is infeasible to derive the user identity ID_i through $h(ID_i)$. Furthermore, the login message is dynamic in each login. Among the parameters of login message, F_i is associated with nonce R_1 and dynamically changed. Consequently, the attacker cannot identify the person who is trying to login.

3.2. Relay Attack

The proposed scheme can resist replay attack because the login request message and the mutual authentication message both contain the nonce instead of timestamp. Suppose that the attacker has intercepted a previous login request message $\{h(ID_i), F_i\}$ from U$_i$, the attacker can resend the same message to S, but he can't continue, because he can't compute $G_i = h(h(ID_i) \oplus x)$ without knowing *x* and can't compute R_1'. For the same reason, the attacker still cannot successfully impersonate the server S to cheat the users by replaying the server's previous mutual authentication message $\{H_i, h(R_1')\}$.

3.3. Impersonation Attack

The proposed scheme can withstand impersonation attack. Assume the attacker intercepts $h(ID_i)$, F_i, H_i, but these information has no meaning to an attacker. He can't derive the secret parameter *x* and password PW_i. Without R_1, R_2, *x* and PW_i, the attacker can't compute H_i, so impersonation can't continue. What's more, the attacker can't impersonation of S, because he can't compute R_1' without knowing the secret key *x*.

3.4. Denial-of-Service Attack

In our proposed scheme, the smart card of user U$_i$ checks the validity of user identity ID_i and password PW_i before update procedure. The attacker has to insert the smart card of user U$_i$ into the smart card reader and has to guess the identity ID_i and password PW_i correctly. Since the smart card computes $A_i' = B_i \oplus h(ID_i \| PW_i)$, $C_i' = h(A_i')$, and compares the computed value of C_i' with the stored value of C_i in its memory to verify the legitimacy of U$_i$ before the smart card accepts the password update request. It is not possible to guess the identity ID_i and password PW_i correctly at the same time in real polynomial time even after getting the smart card of user U$_i$. Therefore, the proposed protocol is secure against DOS attacks.

3.5. Insider Attack

If an attacker obtains B_i and C_i from U$_i$'s smart card, he can't extract sensitive information, like ID_i, PW_i, *x*, because it is computationally infeasible to invert the one-way hash function *h*(). Moreover, he can't extract A_i from B_i without the knowledge of ID_i and PW_i. Furthermore, if the attacker is a legal user U$_i$, he can't obtain x from his smart card. Thus, the insider attack is resisted.

3.6. Password Guessing Attack

In our scheme, U$_i$'s password is only involved with $h(ID_i \| PW_i)$ instead of login request message $\{h(ID_i), F_i\}$ or response message $\{H_i, h(R_1')\}$, it is more difficult for an attacker to compute a valid authentication request message without knowing the server's secret value *x*. Therefore, we believe that the on-line password guessing attacks can be prevented more efficiently.

On the other hand, in our scheme U$_i$'s login message, *i.e.* $h(ID_i)$, F_i, are well-protected and un-involved with U$_i$'s password. This design eliminates the correlation between U$_i$'s password and the transmitted messages, *i.e.* $h(ID_i)$, F_i, H_i, an attacker has no ability to examine his guessed password with previous legitimate request or reply message in an off-line mode. Hence, our scheme is secure against the off-line password guessing attack.

3.7. Stolen Smart Card Attack

Our scheme can prevent stolen smart card attack. If the smart card is stolen or lost, the attacker can extract the secret information B_i and C_i from the smart card. With the parameter, the attacker tries to impersonate the user to login to the server S, however, he must produce a valid login request message $\{h(ID_i), F_i\}$. It can be observed that it is impossible to compute A_i and F_i from the given parameters without knowing *x*, ID_i, and PW_i, so the attacker can't generate a valid login message.

3.8. Parallel Session Attack

Assume the attacker can masquerade as legitimate user U$_i$ by replaying a login request message $\{h(ID_i), F_i\}$.

However, he can't compute the agreed session key $SK = h(R_1 \| R_2)$ between user U_i and server S because he does not know the values of x, R_1, R_2. Therefore, the proposed scheme is secure against parallel session attack.

3.9. Mutual Authentication

Our scheme provides mutual authentication of U_i and S. In our scheme, S sends mutual authentication message $\{H_i, h(R_1')\}$ to U_i validate its authenticity. The value of H_i is calculated by G_i which is only known to U_i and S and this message is infeasible to forge by a fake server to impersonate the S.

3.10. Session Key Agreement

The proposed scheme provides session key agreement during the authentication phase. Suppose the attacker obtains the secret values in the legal user's smart card and intercepts messages communicating between the user and the server, he may attempt to compute the session key *SK*. However, he can't continue without knowing R_1 and R_2.

4. Performance Comparison

In this section, we summarize some performance issues of the proposed scheme. We compare the proposed scheme with related schemes in terms of cost and security requirements.

4.1. Cost Analysis

An efficient authentication scheme must take computation and communication cost into consideration during user's authentication. The computation cost of each phase is defined as the total time of various operations executed in that phase. The communication cost of authentication includes the cost of transmitting messages involved in the authentication scheme. We mainly focus on the computations of registration, login and authentication phases since these phases are the main body of the proposed scheme.

In order to carry out the computation cost evaluation, we use the following notations: T_h and T_s are defined as the execution time of the one-way hash function and symmetric operations. Because exclusive-or operation and concatenation operation require very low execution time, it is usually neglected considering its computational cost. The time complexity associated with the different operations can be expressed as $T_{\oplus} \ll T_h < T_s$. The comparative results are shown in **Table 2**.

From the table, it is noticed that our scheme requires nearly the same computation as other related schemes, but our scheme provides more security.

In addition, we have shown the comparison of communication cost between our scheme and related scheme. The comparative results are shown in **Table 3**, we assume that the output size of secure one-way hash function is 128 bits. For comparison, we also assume that, the lengths of ID_i, PW_i, x, y are 128 bits, and the sizes of timestamps and random number are 64 bits.

From the table, it is noticed that the communication cost of Das *et al*.'s scheme is the least with 448 bits, because, it does not support mutual authentication. However, our scheme needs less bits than others.

4.2. Security Requirements Analysis

In this section, we summarize the security features of our proposed scheme and compare its security robustness with related schemes. The comparative results are shown in **Table 4**.

From the table, it is noticed that our scheme is more secure and robust than other schemes and achieves more security requirements, which were not considered in the their scheme and are essentially required in implementing a practical and universal remote user authentication scheme using smart cards.

5. Conclusions

In this paper, we see that several dynamic user authentication schemes have weaknesses because of using timestamps. Besides, the implement of strict and safe time synchronization is very difficult and increases network overhead. To eliminate these weaknesses, we propose a new dynamic user authentication scheme based on

Table 2. Computation cost comparison.

Scheme	Phase		
	Registration	Login and verification	Total cost
Proposed scheme	$4T_h$	$8T_h$	$12T_h$
Khan *et al*. (2011)	$2T_h$	$10T_h$	$12T_h$
Wang *et al*. (2009)	$2T_h$	$6T_h$	$8T_h$
Das *et al*. (2004)	$2T_h$	$7T_h$	$9T_h$

Table 3. Communication cost comparison.

Scheme	From user to server		From server to user		Total cost
	Message	Cost	Message	Cost	
Proposed scheme	h(ID$_i$), F$_i$	256 bits	$H_i, h(R_i')$	256 bits	512 bits
Khan *et al*. (2011)	CID$_i$, T$_i$, d, C$_i$	384 bits	C2,Ts	192 bits	576 bits
Wang *et al*. (2009)	ID$_i$, CID$_i$, N$_i$,T	448 bits	a', T*	192 bits	640 bits
Das *et al*. (2004)	CID$_i$, N$_i$, C$_i$, T	448 bits	-	-	448b its

Table 4. Security properties comparison.

Scheme	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
Proposed	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Khan *et al.*	N	Y	N	Y	N	Y	Y	N	Y	Y
Wang *et al.*	N	N	N	N	N	N	Y	N	Y	N
Das *et al.*	N	N	N	Y	N	Y	Y	Y	N	N

S1: Resist impersonation attack; S2: Resist DOS attack; S3: Resist insider attack; S4: Resist replay attack; S5: Resist password guessing attack; S6: Resist stolen smart card attack S7: Resist Parallel session attack; S8: Provide user's anonymity; S9: Provide mutual authentication; S10: Provide session key agreement.

nonce instead of timestamps. Mutual authentication is performed using a challenge-response handshake between user and remote server. Moreover, our scheme uses hashing functions to implement user's anonymity and session key agreement. The other merits include: 1) our scheme provides a secure password change method to prevent the adversary from updating password freely; 2) our scheme can resist various attack, including forward secrecy; 3) our scheme requires less computation and communication traffic; 4) it is a nonce-based scheme to avoid the time-synchronization problem.

Therefore, this scheme is well suited to the network-based application systems. In our future work, we would carry on experiments if the conditions are met.

Acknowledgements

Yang Xiaohui, Cui Xinchun, Cao Zhenliang and Hu Ziqiang thank the anonymous reviewers for their valuable comments and suggestions.

References

[1] Lamport, L. (1981) Password Authentication with Insecure Communication. *Communications of the ACM*, **24**, 770-772. http://dx.doi.org/10.1145/358790.358797

[2] Yoon, E.J., Ryu, E.K. and Yoo, K.Y. (2004) Further Improvement of an Efficient Password Based Remote User Authentication Scheme Using Smart Cards. *IEEE Transactions on Consumer Electronics*, **50**, 612-614. http://dx.doi.org/10.1109/TCE.2004.1309437

[3] Tina, X., Zhu, R.W. and Wong, D.S. (2007) Improved Efficient Remote User Authentication Schemes. *International Journal of Network Security*, **4**, 149-154.

[4] Yang, L. and Ma, J.F. (2011) Trusted Mutual Authentication Scheme with Smart Cards and Passwords. *Journal of University of Electronic Science and Technology of China*, **4**, 128-133.

[5] Das, M.L., Saxena, A. and Gulati, P. (2004) A Dynamic Id-Based Remote User Authentication Scheme. *IEEE Trans-*

actions on Consumer Electronics, **50**, 629-631. http://dx.doi.org/10.1109/TCE.2004.1309441

[6] Wang, Y.Y., Liu, J.Y., Xiao, F.X. and Dan, J. (2009) A More Efficient and Secure Dynamic Id-Based Remote User Authentication Scheme. *Computer Communications*, **32**, 583-585. http://dx.doi.org/10.1016/j.comcom.2008.11.008

[7] Khan, M.K., Kim, S.K. and Alghathbar, K. (2011) Cryptanalysis and Security Enhancement of a More Efficient & Secure Dynamic ID-Based Remote User Authentication Scheme. *Computer Communications*, **34**, 305-309. http://dx.doi.org/10.1016/j.comcom.2010.02.011

[8] Kocher, P., Jaffe, J. and Jun, B. (1999) Differential Power Analysis. *Lecture Notes in Computer Science*, **1666**, 388-397.

[9] Messerges, T.S., Dabbish, E.A. and Sloan, R.H. (2002) Examining Smart-Card Security under the Threat of Power Analysis Attacks. *IEEE Transactions on Computers*, **51**, 541-552. http://dx.doi.org/10.1109/TC.2002.1004593