INFORMATICA, 2003, Vol. 14, No. 4, 551–558

Security of Improvement on Methods for Protecting Password Transmission

Chou-Chen YANG
Department of Management Information System, National Chung Hsing University
250 Kuo Kuang Road, 402 Taichung, Taiwan, R.O.C.

Ting-Yi CHANG
Department of Computer and Information Science, National Chiao Tung University
1001 Ta Hsueh Road, Hsinchu, Taiwan, R.O.C.

Min-Shiang HWANG
Department of Management Information System, National Chung Hsing University
250 Kuo Kuang Road, 402 Taichung, Taiwan, R.O.C.
e-mail: mshwang@nchu.edu.tw

Received: June 2003

Abstract. Recently, Tseng *et al.* proposed an improvement on Peyravian and Zunic's protected password transmission scheme and protected changing scheme to remove some security flaws. However, as we will point out in this paper, any adversary can intercept the request for changing the password sent by a legal user and modify it with a wrong password. Furthermore, we shall also propose an improved version of their protected password changing scheme to help it out of the trouble.

Key words: authentication, cryptography, discrete logarithm, password.

1. Introduction

The rapid growth of networks in both number and size encourages more and more computers to link together for sharing various kinds of data and exchanging huge amounts of information. Various resources distributed among the hosts are shared across the network in the form of network services provided by servers. Before providing such services, servers should have the ability to authenticate the users. Otherwise, any adversary can easily impersonate a legal user to login the server. Thus, user authentication is one of the most important security services for secure communication.

Password-based authentication schemes are the most widely used methods for user authentication since they allow people to choose their own passwords without any devices. In practice, people hardly find long random string passwords easy to use and remember. It would be much more user-friendly if the password is a meaningful string that people can recognize easily such as a natural language phrase. However, the easy-to-remember passwords narrow down the possibilities and make it easier for attackers to succeed. Most of such schemes (Botting, 1997; Horng, 1995; Hwang, 1999; Hwang *et al.*, 2001; Jablon, 1996; Lee *et al.*, 2002; Li *et al.*, 2001) require symmetric or asymmetric cryptosystems to encrypt the passwords before traveling them over public networks. In 2000, Peyravian and Zunic (Peyravian and Zunic, 2000) proposed a pair of schemes that allow users to transmit passwords over public networks and arbitrarily change their passwords without employing any symmetric cryptosystem (such as DES (Smid, 1988), Rindael (Daemen and Rijmen, 2001), and others (Hwang, 2000; Hwang *et al.*, 2002b)) or asymmetric cryptosystem (such as RSA (Changchien *et al.*, 2002; Hwang, 2000; Rivest *et al.*, 1978), ElGamal (ElGamal, 1985; Hwang *et al.*, 2002a), etc.).

Unfortunately, Tseng *et al.* (2001)pointed out that Peyravian and Zunic's schemes are vulnerable to such attacks as the guessing attack and server spoofing. They proposed improved schemes based on the Diffie-Hellman key agreement scheme (Diffie and Hellman, 1976) to withstand these attacks. At the same time, Hwang and Yeh (Hwang and Yeh, 2002) also pointed out that those attacks threaten the security of Peyravian and Zunic's schemes and also repaired the security flaws in their schemes based on the server's asymmetric cryptosystem. However, to avoid the adversary from replacing the original public keys of the servers with her/his own, certificates (e.g., digital signatures) should be introduced in Hwang and Yeh's schemes. A certificate from a trusted authority is what servers with public keys will ask for before providing services. This means users need large storage spaces for certificates hard to remember, and more bandwidth is also needed for verifying the signatures.

In this paper, we will point out Tseng *et al.*'s protected password changing scheme is vulnerable to the modification attack; that is, any adversary can intercept the request for changing passwords sent by a legal user and modify it with a wrong password. At the same time, we will propose an improvement on their scheme without using any symmetric or asymmetric cryptosystems.

The organization of the paper is as follows. In the next section, we will briefly review Tseng *et al.*'s protected password changing scheme, and the modification attack will also be introduced. In Section 3, we will propose our improved version to enhance the security of their scheme. In Section 4, we will analyze the security of our improved scheme. Finally, we shall conclude this paper with Section 5.

2. Modification Attack on Tseng *et al.*'s Scheme

In this section, we shall first briefly review Tseng *et al.*'s protected password changing scheme and then show how the modification attack can work on their scheme. In the system, the server publishes two large prime numbers p and q such that $q|p-1$. Let g be a generator with order q in the Galois field $GF(p)$, which is based on the Diffie-Hellman assumption. The client (user) has the identity id and the corresponding password pw. The password pw is only secretly known to both the client and the server. The server employs a one-way hash function $H(\cdot)$ to store id and pw as $verify\_idpw\_digest = H(id, pw)$ in the database. The protected password changing scheme works as follows:

Step 1. Client ⟶ Server: id, C_idpw_digest
The client chooses a random number $c \in [1, q-1]$ and computes $rc = g^c \bmod p$. Then, it computes $C\_idpw\_digest = H(id, pw) \oplus rc$ ($\oplus$ denotes the exclusive operator) and sends it along with id to the server.

Step 2. Server ⟶ Client: $S\_idpw\_digest, S\_auth\_token$
The server first recovers rc from C_idpw_digest by computing $C\_idpw\_digest \oplus verify\_idpw\_digest$. Then, the server chooses a random number $s \in [1, q-1]$ and computes $rs = g^s \bmod p$ and $rcs = (rc)^s = g^{cs} \bmod p$. Next, the server computes $S\_idpw\_digest = verify\_idpw\_digest \oplus rs$ and $S\_auth\_token = H(verify\_idpw\_digest, rcs, rc)$, and then it sends them to the client.

Step 3. Client ⟶ Server: $id, C\_auth\_token, C\_new\_idpw\_digest$
The client first recovers rs from S_idpw_digest by computing $S\_idpw\_digest \oplus H(id, pw)$. Then it computes $rcs = (rs)^c = g^{sc} \bmod p$ and uses it together with its own $H(id, pw)$ and rc to compute $H(H(id, pw), rcs, rc)$, which is then compared with the received S_auth_token from the server. If they match, the client computes $C\_auth\_token = H(H(id, pw), rcs, rs)$. Then, the client chooses a new password new_pw and computes $C\_new\_idpw\_digest = H(id, new\_pw) \oplus H(H(id, pw), rcs)$. Finally, the client sends $\{id, C\_auth\_\ token, C\_new\_idpw\_digest\}$ to the server.

Step 4. The server uses its own $verify\_idpw\_digest$, rcs and rs to compute $H(verify\_idpw\_digest, rcs, rs)$ and compares it with the received C_auth_token. If they match, the server recovers $H(id, new\_pw)$ from $C\_new\_\ idpw\_digest$ by computing $C\_new\_idpw\_digest \oplus H(verify\_idpw\_digest, rcs)$ and then stores $verify\_idpw\_ digest = H(id, new\_pw)$ in the database.

The difference between the protected password transmission scheme and protected password changing scheme is that the client additionally sends $C\_new\_idpw\_digest$ to the server for changing passwords in the latter scheme.

Note that the client sends $\{id, C\_auth\_token, C\_new\_idpw\_digest\}$ in Step 3 to the server, and the messages C_auth_token and $C\_new\_idpw\_digest$ are used to enable the server to authenticate the client and to obtain $H(id, new\_pw)$, respectively. However, because the two messages are separated, the attacker can replace $C\_new\_idpw\_digest$ with a random number ra. After receiving $\{id, C\_auth\_token, ra\}$, the server checks the validity of C_auth_token. Since C_auth_token is generated by the legal client, the server will accept it. Then, the server computes $ra \oplus H(verify\_idpw\_digest, rcs)$ and stores $verify\_idpw\_digest = ra \oplus H(verify\_idpw\_digest, rcs)$ in the database.

Unfortunately, the client is mistakenly convinced that it has successfully changed from the old password pw to a new password new_pw. When the client tries to login the server the next time, the server will reject the client's login request because the client cannot recover rs from S_idpw_digest by computing $S\_idpw\_digest \oplus H(id, new\_pw)$ and therefore cannot compute C_auth_token correctly. As a result, the server will conclude that the client is illegal, and the client will not be able to change its password successfully.

3. The Improved Scheme

To get rid of the previously shown security flaw, we shall propose an improved protected password-changing scheme as follows:

Step 1. Client ⟶ Server: $id, C\_idpw\_digest, C\_new\_idpw\_digest$
The client chooses a random number $c \in [1, q-1]$ and computes $rc = g^c \bmod p$. Then, the client chooses a new password new_pw and uses rc to compute $C\_idpw\_digest = H(id, pw) \oplus rc$ and $C\_new\_idpw\_digest = H(id, new\_pw) \oplus rc$. The client sends $\{id, C\_idpw\_digest, C\_new\_idpw\_\ digest\}$ to the server.

Step 2. Server ⟶ Client: $S\_idpw\_digest, S\_auth\_token$
The server first recovers rc from C_idpw_digest by computing $C\_idpw\_digest \oplus verify\_idpw\_digest$. Then, the server uses the recovered rc to obtain $H(id, new\_pw)$ from $C\_new\_idpw\_digest$ by computing $C\_new\_idpw\_digest \oplus rc$. Then, the server chooses a random number $s \in [1, q-1]$ and computes $rs = g^s \bmod p$ and $rcs = (rc)^s = g^{cs} \bmod p$. Then, the server uses its own rs, rcs and the recovered rc to compute $S\_idpw\_digest = verify\_idpw\_digest \oplus rs$ and $S\_auth\_token = H(verify\_idpw\_digest, rcs, rc)$. The server sends $\{S\_idpw\_digest, S\_auth\_token\}$ to the client.

Step 3. Client ⟶ Server: id, C_auth_token
The client first recovers rs from S_idpw_digest by computing $S\_idpw\_digest \oplus H(id, pw)$. Then it computes $rcs = (rs)^c = g^{sc} \bmod p$ and uses it together with its own $H(id, pw)$ and rc to compute $H(H(id, pw), rcs, rc)$, which is then compared with the received S_auth_token from the server. If they hold, the server is authenticated and the client uses its own rcs, $H(id, new\_pw)$ and the recovered rs to compute $C\_auth\_token = H(H(id, pw), rcs, rs) \oplus H(id, new\_pw)$. Finally, the client sends $\{id, C\_auth\_token\}$ to the server.

Step 4. The server uses its own $verify\_idpw\_digest$, rcs and rs to compute $H(verify\_idpw\_digest, rcs, rs)$. Then, the server uses the recovered $H(id, new\_pw)$ in the Step 1 to compute $C\_auth\_token \oplus H(id, new\_pw)$ and compares the result with $H(verify\_idpw\_digest, rcs, rs)$. If they hold, the server stores $verify\_idpw\_digest = H(id, new\_pw)$ in the database.

Obviously, our improved scheme does not employ any symmetric or asymmetric cryptosystems and does not increase the client's or server's computational complexity. Without sending $C\_new\_idpw\_digest$ and $C\_auth\_token = H(H(id, pw), rcs, rs)$, the scheme is the same as Tseng *et al.*'s protected password transmission scheme.

4. Security Analysis

In this section, we shall first list the security definitions, and then several possible attacks will be analyzed to demonstrate the security of our improved scheme.

DEFINITION 1. A weak secret (password) is a value of low entropy $w(k)$, which can be guessed in a polynomial time.

DEFINITION 2. The Diffie-Hellman assumption is that given $g^c \bmod p$ and $g^s \bmod p$, to compute $g^{cs} \bmod p$ is hard.

DEFINITION 3. A secure one-way hash function $Y = H(X)$ is one where given X to compute Y is easy and given Y to compute X is hard.

Replay attacks

The attacker intercepts $\{id, C\_idpw\_digest, C\_new\_idpw\_digest\}$ sent by the client in Step 1 and uses it to masquerade as the client to send the login request to the server. However, he/she has no ability to make a correct response C_auth_token in Step 3 because the random number rs generated by the server is different every time. On the other hand, since the message sent by the server and the client is different, the attacker cannot intercept any message between them and then replay it to the other party.

Guessing attacks

Ding and Horster (Ding and Horster, 1995) divided password-guessing attacks into three classes: (1) Detectable on-line guessing attacks, (2) Undetectable on-line guessing attacks, and (3) Off-line guessing attacks. Because the detectable on-line guessing attacks can be prevented by letting the server take appropriate intervals between trials, we shall only separately show that the latter two guessing attacks will fail in our scheme as follows.

Undetectable on-line guessing attacks

The attacker attempts to use a guessed password pw' in an on-line transaction. She/he verifies the correctness of her/his guess by observing the response of the server. A failed guess cannot be detected by the server. The attacker disguises as the client and cheats the server. She/he can generate $C\_idpw\_digest' = H(id, pw') \oplus (ra = g^a \bmod p)$ and $C\_new\_idpw\_digest' = ra \oplus H(id, new\_\ pw')$ by herself/himself and get $S\_idpw\_digest = verify\_idpw\_digest \oplus rs$, $S\_auth\_token = H(verify\_idpw\_digest, (H(id, pw') \oplus ra \oplus H(id, pw))^b \bmod p, H(id, pw') \oplus ra \oplus H(id, pw))$ sent from the server in Step 2. In order to verify S_auth_token, the attacker must control $ras = g^{as} \bmod p$ by finding $rs = rs \oplus H(id, pw) \oplus H(id, pw')$ and $ra = ra \oplus H(id, pw) \oplus H(id, pw')$ simultaneously. However, the probability is $2^{-w(k)}$. Even if pw' is guessed, the attacker cannot verity S_auth_token and reply the correct C_auth_token to the server. On the other hand, the attacker still cannot mount the on-line guessing attacks for guessing new_pw because no message from the server includes new_pw. Therefore, no undetectable on-line guessing attacks can do any harm to our scheme.

Off-line guessing attacks

Since easy-to-remember passwords do not have enough entropy, the attacker can guess the password have off-line, independent of the server. The attacker has the knowledge of C_idpw_digest, S_idpw_digest, C_auth_token, and S_auth_token. He/she first guesses a password pw' and then finds $rc = rc \oplus H(id, pw) \oplus H(id, pw')$ and $rs = rs \oplus H(id, pw) \oplus H(id, pw')$. However, the adversary has to break the Diffie-Hellman assumption to find rcs and then verify his/her guess. Without knowing rc, rs and rcs, the attacker cannot guess new_pw from $C\_new\_idpw\_digest$ in Step 1 and C_auth_token in Step 3.

Server spoofing

Only the real server has the ability to recover rc from C_idpw_digest and then make a correct response S_auth_token in Step 2. An illegal client cannot recover rs from S_idpw_digest and then make a correct response C_auth_token in Step 3. The improved scheme can also achieve the property of mutual authentication.

Modification attack

The attacker tries to make the client convinced that it has successfully changed the password from pw to new_pw. In fact, the $verify\_idpw\_digest$ stored in the server's database is a random number. The attacker modifies $C\_new\_idpw\_digest$ to a random number ra in Step 1. After receiving $\{id, C\_idpw\_digest, ra\}$, the server recovers rc and uses it to obtain $ra \oplus rc$ and then sends $\{S\_idpw\_digest, S\_auth\_token\}$ to the client in Step 2. After receiving $\{S\_idpw\_digest, S\_auth\_token\}$, the client first authenticates the server and then sends $\{id, C\_auth\_token\}$ to the server in Step 3. After receiving $\{id, C\_auth\_token\}$, the server first uses the recovered $ra \oplus rc$ to compute $C\_auth\_token \oplus (ra \oplus rc)$ and then compares it with $H(verify\_idpw\_digest, rcs, rs)$. Obviously, $C\_auth\_token \oplus (ra \oplus rc)$ is not equal to $H(verify\_idpw\_digest, rcs, rs)$ because $C\_auth\_token \oplus (ra \oplus rc) = H(H(id, pw), rcs, rs) \oplus H(id, new\_pw) \oplus (ra \oplus rc)$. However, even if the attacker can compute $H(H(id, pw), rs, rcs) \oplus (ra \oplus rc)$ as C_auth_token and send it to the server in Step 3, she/he cannot break the Diffie-Hellman assumption to obtain rc, rs and rcs, and get $H(id, pw)$.

After the above discussions, we have come to the conclusion that our improved scheme can withstand replay attacks, guessing attacks, and server spoofing, and it can work successfully against modification attacks. Moreover, a session key between the client and the server can be established as $H(rcs)$. When the password is compromised, the attacker still cannot reveal an old session key. The scheme can still live up to the requirement of perfect forward secrecy (Jablon, 1996).

5. Conclusion

In this paper, we have shown that Tseng *et al.*'s protected password changing scheme is vulnerable to the modification attack. In addition, we have proposed our improved scheme to effectively mend the security flaw without employing any symmetric or asymmetric cryptosystems.

Acknowledgements

The authors wish to thank many anonymous referees for their suggestions to improve this paper. Part of this research was supported by the National Science Council, Taiwan, R.O.C., under contract no. NSC90-2213-E-324-004.

References

Botting, J. (1997). Security on the Internet: authenticating the use. *Telecommunications*, **31**(12), 77–80.

Changchien, S.W., M.S. Hwang and K.F. Hwang (2002). A batch verifying and detecting multiple RSA digital signatures. *International Journal of Computational and Numerical Analysis and Applications*, **2**(3), 303–307.

Daemen, J., and V. Rijmen (2001). Rijndael, the advanced encryption standard. *Dr. Dobb's Journal*, **26**(3), 137–139.

Diffie, W., and M. Hellman (1976). New direction in cryptography. *IEEE Transactions On Information Theory*, **22**(6), 472–492.

Ding, Y., and P. Horster (1995). Undetected on-line password guessing attacks. *ACM Operating Systems Review*, **29**(4), 77–86.

ElGamal, T. (1985). A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, **IT-31**, 469–472.

Horng, G. (1995). Password authentication without using password table. *Information Processing Letters*, **55**(4), 247–250.

Hwang, J.J., and T.C. Yeh (2002). Improvement on Peyravian-Zunic's password authentication schemes. *IEICE Trans. on Communications*, **E85-B**(4), 823–825.

Hwang, M.S. (1999). A remote password authentication scheme based on the digital signature method. *International Journal of Computer Mathematics*, **70**, 657–666.

Hwang, M.S. (2000). A new redundancy reducing cipher. *Informatica*, **11**(4), 435–440.

Hwang, M.S., C.C. Chang and K.F. Hwang (2002a). An ElGamal-like cryptosystem for enciphering large messages. *IEEE Transactions on Knowledge and Data Engineering*, **14**(2), 445–446.

Hwang, M.S., S.W. Changchien and C.C. Lee (2002b). A new method to strengthen ciphers. *International Journal of Pure and Applied Mathematics*, **3**(2), 187–192.

Hwang, M.S., C.C. Lee and Y.L. Tang (2001). An improvement of SPLICE/AS in WIDE against guessing attack. *Informatica*, **12**(2), 297–302.

Hwang, M.S., I.C. Lin and K.F. Hwang (2000). Cryptanalysis of the batch verifying multiple RSA digital signatures. *Informatica*, **11**(1), 15–18.

Jablon, D.P. (1996). Strong password only authenticated key exchange. *Computer Communication Review*, **26**, 5–26.

Lee, C.C., M.S. Hwang and W.P. Yang (2002). A flexible remote user authentication scheme using smart cards. *ACM Operating Systems Review*, **36**(3), 46–52.

Li, L.H., I.C. Lin and M.S. Hwang (2001). A remote password authentication scheme for multi-server architecture using neural networks. *IEEE Transactions on Neural Networks*, **12**(6), 1498–1504.

Peyravian, M., and N. Zunic (2000). Methods for protecting password transmission. *Computers and Security*, **19**(5), 466–469.

Rivest, R.L., A. Shamir and L. Adleman (1978). A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM*, **21**, 120–126.

Smid, M.E., and D. K. Branstad (1988). The data encryption standard: past and future. *Proc. of the IEEE*, **76**, 550–559.

Tseng, Y.M., J.K. Jan and H.Y. Chien (2001). On the security of methods for protecting password transmission. *Informatica*, **12**(3), 469–477.

Ch.-Ch. Yang received his BS in industrial education from the National Kaohsiung Normal University, in 1980, and his MS in electronic technology from the Pittsburg State University, in 1986, and his PhD in computer science from the University of North Texas, in 1994. He has been an associate professor in the Dept. of Information and Communication Engineering since 1994. His current research interests include network security, mobile computing, and distributed system.

T.-Y. Chang received the BS in information management from Chaoyang University of Technology (CYUT), Taichung, Taiwan, Republic of China, in 2001, and his MS in Department and Graduate Institute of Computer Science and Information Engineering from CYUT, in 2003. He is currently pursuing his PhD in computer and information science from National Chiao Tung University, Taiwan, Republic of China. His current research interests include information security, cryptography, and mobile communications.

M.-Sh. Hwang was born on August 27, 1960 in Tainan, Taiwan, Republic of China (ROC.). He received the BS in electronic engineering from National Taipei Institute of Technology, Taipei, Taiwan, ROC, in 1980; the MS in industrial engineering from National Tsing Hua University, Taiwan, in 1988; and the PhD in computer and information science from National Chiao Tung University, Taiwan, in 1995. He also studied applied mathematics at National Cheng Kung University, Taiwan, from 1984–1986. Dr. Hwang passed the national higher examination in field “Electronic Engineer”in 1988. He also passed the national telecommunication special examination in field “Information Engineering”, qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999–2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002–2003. He obtained the 1997, 1998, 1999, 2000, and 2001 distinguished research awards of the National Science Council of the Republic of China. He is currently a professor of the Department of Management Information System, National Chung Hsing University, Taiwan, ROC. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang had published 80 articles on the above research fields in international journals.

Pagerintų slaptažodžių perdavimo apsaugos metodų saugumas

Chou-Chen YANG, Ting-Yi CHANG, Min-Shiang HWANG

Neseniai Tseng *et al.* pasiūlė pagerintą Peyravian ir Zunic saugaus slaptažodžių perdavimo ir pakeitimo schemas, kurios išvengia saugumo problemų. Tačiau, šiame straipsnyje mes parodysime, kad kiekvienas pažeidėjas gali perimti legalaus vartotojo siunčiamą slaptažodžio pakeitimo užklausą ir modifikuoti ją naudojant neteisingą slaptažodį. Be to, mes siūlome pagerintą saugaus slaptažodžio pakeitimo schemą, kuri apsaugo nuo minėtos saugumo problemos.