On the Security of Some Password Authentication Protocols

Bin-Tsan HSIEH
Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan 701
e-mail: bintsan@csi.ncku.edu.tw

Hung-Min SUN
Department of Computer Science , National Cheng Kung University
Hsinchu, Taiwan 300
e-mail: hmsun@cs.nthu.edu.tw

Tzonelih HWANG
Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan 701

Received: November 2002

Abstract. In an internet environment, such as UNIX, a remote user has to obtain the access right from a server before doing any job. The procedure of obtaining acess right is called a user authentication protocol. User authentication via user memorable password provides convenience without needing any auxiliary devices, such as smart card. A user authentication protocol via username and password should basically withstand the off-line password guessing attack, the stolen verifier attack, and the DoS attack. Recently, Peyravian and Zunic proposed one password transmission protocol and one password change protocol. Later, Tseng *et al.* (2001) pointed out that Peyravian and Zunic's protocols can not withstand the off-line password guessing attack, and therefore proposed an improved protocol to defeat the attack. Independently, Hwang and Yeh also showed that Peyravian and Zunic's protocols suffer from some secury flaws, and an improved protocol was also presented. In this paper, we show that both Peyravian and Zunic's protocols and Tseng *et al.*'s improved protocol are insecure against the stolen verifier attack. Moreover, we show that all Peyravian and Zunic's, Tseng *et al.*'s, and Hwang and Yeh's protocols are insecure against DoS attack.

Key words: computer security, network security, protocol, cryptanalysis, password, authentication, hash function, cryptography.

1. Introduction

In an internet environment, a remote user has to obtain the access right from a server, such as a UNIX workstation, before doing any job. The procedure of obtaining acess right is called a user authentication protocol. It is very common that a server in a network of resources is used to provide controlled access to the network or to applications residing within the network. Therefore, it is necessary for the server to authenticate the client via username and password. User authentication via user memorable password provides convenience without needing any auxiliary devices, such as smart card.

Recently, Peyravian and Zunic (2000) proposed a secure protocol to solve the above problem without revealing passwords over untrusted networks. In addition, they also presented a secure protocol for changing an old password to a new password. On the other hand, people often tend to choose easy-to-remember passwords (or referred to as "weak passwords"), which are vulnerable to the password guessing attack (or referred to as "dictionary attack") if some verifiable information for password is provided. Hence it is very essential for password-based protocols, e.g., (Jablon, 1996), to defeat the off-line password guessing attack (Bellovin and Merrit, 1992) (it is natural that the on-line password guessing attack can not be defeated by means of protocols themselves). Furthermore, some password-based protocols keep verifiers, the hashed images of passwords, in the server's database instead of storing plain passwords such that any stolen verifier cannot make the impersonation for the client succeed (it is natural that the impersonation for the server can succeed due to stolen verifiers).

Another kind of modern attacks is the DoS attack. The DoS attack stands for Denial-of-Service which is an attack leading a legal user can not login the server or the server can not provide service normally. Although any malicious benavior of an attacker can exhaust server's resources, such as CPU time, or disrupt the server easily, however, a robust user authentication protocol should prevent from the DoS attack by strict integrity check.

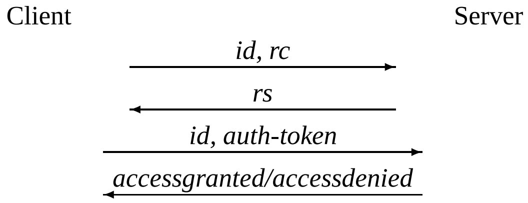
Tseng *et al.* (2001) pointed out that Peyravian and Zunic's protocols can not withstand the off-line password guessing attack, and therefore proposed an improved protocol to defeat the attack. Independently, Hwang and Yeh also showed that Peyravian and Zunic's protocols suffer from some secury flaws, and an improved protocol was also presented. In this paper, we show that both Peyravian and Zunic's protocols and Tseng *et al.*'s improved protocol are insecure against the stolen verifier attack. Moreover, we show that all Peyravian and Zunic's, Tseng *et al.*'s, and Hwang and Yeh's protocols are insecure against the DoS attack.

The remainder of this paper is organized as follows. In Section 2, we first review protocols proposed by Peyravian and Zunic. Section 3 gives Tseng *et al.*'s attack and their improved protocol. Section 4 gives Hwang and Yeh's attacks and their improved protocol. In Section 5, we show that Peyravian and Zunic's protocols and Tseng *et al.*'s improved protocol are insecure against the stolen verifier attack. Furthermore, we also point out that all Peyravian and Zunic's, Tseng *et al.*'s and Hwang and Yeh's protocols can not withstand the DoS attack. Section 6 gives our conclusions.

2. Peyravian and Zunic's Protocols Initialization

The server computes and stores the hashed image of the user's password *(pw)* and identity *(id)* as follows: *idpw-digest = H(id,pw)*, where *H()* is a public one-way hash function.

2.1. *Protected Password Transmission Protocol*



Step 1. The client chooses a random number *rc*, and then sends (*id*, *rc*) to the server.

Step 2. The server chooses a random number *rs*, and then sends it to the client.

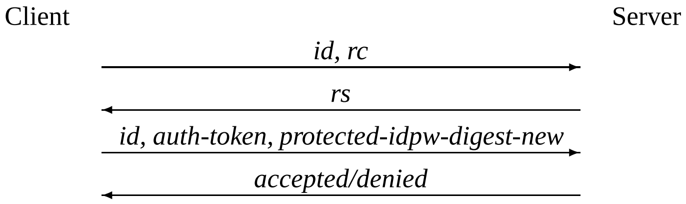
Step 3. The client computes *idpw-digest*$=$*H*(*id*, *pw*), and then computes *auth-token*$=$ *H*(*idpv-digest*, *rc*, *rs*) and sends (*id*, *auth-token*) to the server.

Step 4. The server verifies the validity of the received *auth-token*. If it is valid, the server sends a message to the client giving him permission to access the server. Otherwise, it sends a message to the client denying him permission to access the server.

In general, the user may want to change his password from time to time. To provide the flexibility, Peyravian and Zunic also proposed the following protocol for the user to change his password.

2.2. *Protected Password Change Protocol*

Assume that the user wants to change his current password (*pw*) to a new password (*new-pw*). Peyravian and Zunic's protocol works as follows:



Step (a). The client chooses a random number *rc*, and then sends (*id*, *rc*) to the server.

Step (b). The server chooses a random number *rs* and sends it to the client.

Step (c). The client computes *auth-token* as mat in the previous protocol and *protected-idpw-digest-new* as follows:

idpw-digest-new$=$*H*(*id*, *new-pw*)
auth-token-mask$=$*H*(*idpw-digest*, *rc*$+1$, *rs*)
protected-idpw-digest-new$=$*idpw-digest-new* XOR *auth-token-mask*.

The client sends *id, auth-token*, *protected-idpw-digest-new* to the server.

Step (d). After receiving the message from *Step* (c), the server first verifies *auth-token*. If it passes, the server computes *auth-loken-mask*$=$*H*(*idpw-digest*, *rc*$+1$, *rs*) and *idpw-digest-new*$=$*protected-idpw-digest-new* XOR *auth-token-mask*. Finally, the server updates the user's verifier by *idpw-digest-new* and sends a message to the client accepting the password change.

3. Tseng *et al.*'s Attack and their Improvemed Protocol

Generally, protocols for password authentication or for changing password must be able to defeat the password guessing attack and the stolen verifier attack. Tseng *et al.* (2001) showed that these two protocols, proposed by Peyravian and Zunic, are vulnerable to the off-line password guessing attack, and hence proposed an improved protocol to defeat it.

3.1. *The Off-Line Password Guessing Attack on Peyravian and Zunic's Protocols*

In these two protocols, an adversary who intercepts the information flow during these protocols can mount the off-line password guessing attack suc cessfully. This is because one guess on the password can be verified by check ing whether *auth-token*=*H*(*H*(*id*, *pw*), *rc*, *rs*), where *auth-token*, *id*, *rc*, and *rs* are known to the adversary and *H*() is public.

3.2. *Tseng et al.'s Improved Protocol Initialization*

The server computes and stores the hashed image of the user's password (*pw*) and identity (*id*) as follows: *idpw-digest*=*H*(*id*, *pw*), where *H*() is a public one-way hash function. Server chooses and publishes two large prime numbers p and q such that q divides $p-1$. Let g be a generator with order q in the Galois field $GF(p)$. Their improved protocol works as follows:

Client Server

rc=$g^a \bmod p$
idpw_digest=*H*(*id*, *pw*)
C_idpw_digest
=*idpw_digest*⊕*rc*

id, *C_idpw_digest* $\longrightarrow$

rs=$g^b \bmod p$
rcs=$(rc)^b \bmod p$
S_idpw_digest
=*idpw_digest*⊕*rs*
S_auth_token
=*H*(*idpw_digest*, *rc*, *rcs*)

S_idpw_digest,
S_auth_token $\longleftarrow$

verify *S_auth_token*
C_auth_token
=*H*(*idpw_digest*, *rs*, *rcs*)
idpw_digest_new
=*H*(*id*, *new_pw*)
auth_token_mask
=*H*(*idpw_digest*, *rcs*)

p_idpw_digest_new
=*idpw_digest_new*
⊕*auth_token_mask*

id, C_auth_token,
p_idpw_digest_new
$\xrightarrow{\hspace{6em}}$

verify *C_auth_token*
compute *auth_token_mask*
obtain *idpw_digest_new*

Step (t1). The client chooses a random number a in Z_q^*. Then he computes *rc*=*g*^{a} mod p, *idpw_digest*=*H*(*id, pw*), and *C_idpw_digest*=*idpw_digest*⊕*rc*. Client sends his identity *id* and *C_idpw_digest* to server.

Step (t2). The server chooses a random number b in Z_q^* and computes *rs*=*g*^{b}mod p, *rcs*=(*rc*)^b mod p, *S_idpw_digest*=*idpw_digest*⊕*rs*, and *S_auth_token*=*H*(*idpw_digest*, *rc*, *rcs*). Then server sends *S_idpw_digest* and *S_auth_token* to client.

Step (t3). The client computes *rs*=*S_idpw_digest*⊕*idpw_digest* and *rcs*=(*rs*)^a mod p. Then he verifies *S_auth_token* received from server. If it holds, the server is authenticated. Then the client computes the four values:

C_auth_token=*H*(*idpw_digest*, *rs*, *rcs*)
idpw_digest_new=*H*(*id*, *new_pw*)
auth_token_mask=*H*(*idpw_digest*, *rcs*)
p_idpw_digest_new=*idpw_digest_new*⊕*auth_token_mask*

Then client sends *id*, *C_auth_token*, and *p_idpw_digest_new* to server.

Step (t4). The server verifies *C_auth_token* received from client. If it holds, server computes *auth_token_mask* and retrieves *idpw_digest_new* by computing *p_idpw_digest_new*⊕*auth_token_mask*. Finally server updates *idpw_digest_new*.

4. Hwang and Yeh's Attack and Their Improved Protocol

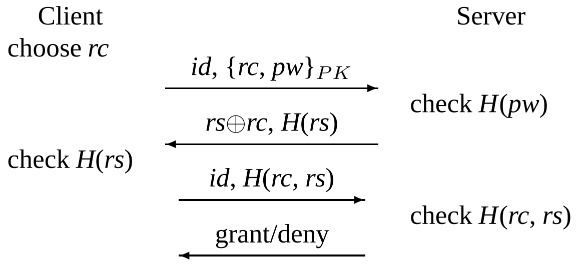
In this section, we review Hwang and Yeh's attacks and their improved protocol as follows.

4.1. *Hwang and Yeh's Attack*

Hwang and Yeh pointed out that Peyravian and Zunic's protocol is vulnerable to the offline password guessing attack. They also showed Peyravian and Zunic's protocol does not authenticate server since any one can play the server's role in the protocol. We omit the detailed descriptions of these attacks here because they are very straightforward.

4.2. *Hwang and Yeh's Improved Protocol*

In Hwang and Yeh's improved protocol, they employ public key infrastructure to overcome the flaws they pointed out. On server side, the server stores user's verifiable information *H*(*pw*). We depict their protocol as follows.



Step (h1). The client chooses a random number *rc*, encrypts *rc*, *pw* using server's public key, and send *id*, {*rc*, *pw*}$_{PK}$ to the server.

Step (h2). The server obtains *rc* and *pw* by decrypting {*rc*, *pw*}$_{PK}$ then computes *H*(*pw*) and chekcs it with the stored *H*(*pw*). If it holds, the server chooses *rs* and sends *rs*⊕*rc*, *H*(*rs*) to the client.

Step (h3). The client computes *rs*=*rs*⊕*rc*⊕*rc* and *H*(*rs*) then checks it with the received *H*(*rs*). If it holds, the client sends *id*, *H*(*rc*, *rs*) to the server.

Step (h4). The server computes and checks *H*(*rc*, *rs*). If it holds, the server sends message "grant" to the client otherwise he sends "deny" to the client.

When the client wants the change his password, he can additionally send *H*(*new_pw*) ⊕*H*(*rc*+1, *rs*) in *Step* (h3).

5. Security Analysis

5.1. *The Stolen Verifier Attack on Peyravian and Zunic's Protocols and Tseng et al.'s Improved Protocol*

It is usually expected that a verifier-based pro tocol can defeat the stolen verifier attack. That is, if the user's verifier is stolen (or known) by an adversary, the adversary can not directly impersonate the user with out running the password guessing attack (Bellare and Rogaway, 2000; Bellovin and Merritt, 1993; Boyko *et al.*, 2000; Kwon and Song, 1999; Wu, 1998). However, in Peyravian and Zunic's password transmission protocol, if the user's verifier *H*(*id*, *pw*) is compromised, even without running the password guessing attack, an adversary can easily impersonate the user by randomly selecting *rc* in *Step* 1 and computing *auth-token*=*H*(*H*(*id*, *pw*), *re*, *rs*) in *Step* 3. Thus, the adversary can successfully impersonate the user to obtain the grant of the server.

In Tseng *et al.*'s improved protocol, if the user's verifier *H*(*id*, *pw*) is compromised, an adversary can first chooses a, then he computes *rc* and *C_idpw_digest* in *Step* t1. Next, he obtains *rs* and *rcs* by computing *S_idpw_digest*⊕*idpw_digest* and (*rs*)^{a} mod p. Then the adversary is able to compute *C_auth_token*, *idpw_digest_new*, *auth_token_mask*, and *p_idpw_digest_new* in *Step* t3. Thus, the adversary can successfully impersonate other users to obtain the grant of the server or change other user's password easily.

Table 1

Security Comparison among Peyravian and Zunic's, Tseng *et al.*'s, and Hwang and Yeh's protocols

	Peyravian and Zunic	Tseng *et al.*	Hwang and Yeh
Public Key Infrastructure	No Need	No Need	Need
Exponential Computation	No Need	Need	No Need
Password Guessing Attack	Insecure	Secure	Secure
Stolen Verifier Attack	Insecure	Insecure	Secure
Denial of Service Attack	Insecure	Insecure	Insecure

5.2. *The DoS Attack on Peyravian and Zunic's, Tseng et al.'s, and Hwang and Yeh's Protocols*

The DoS attack stands for Denial-of-Service which is an attack leading a legal user can not login the server or the server can not provide service normally.

In Peyravian and Zunic's password change protocol, an adversary can change the user's password by randomly selecting *rc* in *Step* (a) and computing *protected_idpw _digest_new* with a new chosen password in *Step* (c). Thus, the adversary can arbitrarily change the user's password. Such a change results in that the user can not correctly login later but the adversary can.

In Tseng *et al.*'s protocol, an attacker can replace *p_idpw_digest_new* by a random number and keep others unchanged in the password change protocol. Thus, the server will update the user's new password as a random number because the unchanged *auth_token_mask* can pass server's check.

In fact, Hwang and Yeh's improved protocol also suffers the same flaw as presented in Tseng *et al.*'s protocol. The attacker can replace $H(n\text{ew\_pw})\oplus H(rc+1, rs)$ with a random number and keep $H(rc, rs)$ unchanged. In other words, the same attack can be applied to Hwang and Yeh's improved protocol directly. We organized comparison table of these three protocols in Table 1.

6. Conclusions

In general, the verifier-based password authentication protocol is widely used in modern systems instead of traditional password authentication protocol. The verifier-based authentication mechanism possesses the advantage of consuming attacker's time in password guessing when the verifier is stolen. A sound verifier-based user authentication protocol should basically withstand the password guessing attack and the stolen verifier attack. Moreover, it also has to defeat the DoS attack since such attack can be performed easily and causes system abnormally. Thus, a user authentication protocol which can not withstand these attacks is not qualified to be used in any system for user authentication.

In Peyravian and Zunic's protocol, they are trying to employ hash function to protect the password transmission without using public key infrastructure or complex computations, such as exponential computation. However, based on our observation, designing a password authentication protocol using only hash funcation is almost impossible to defeat all well-known attacks, such as off-line password attack, stolen verifier attack, and DoS attack.

In this paper, we have shown that both Peyravian and Zunic's protocols and Tseng *et al.*'s improved protocol are insecure against the stolen verifier attack. Moreover, all Peyravian and Zunic's, Tseng *et al.*'s, and Hwang and Yeh's protocols are insecure against the DoS attack.

References

BelIare, M., and P. Rogaway (2000). The authA protocol for password-based authenticated key exchange. *Contribution to the IEEE P1363 Study Group for Future PKC Standards*.

Bellovin, S.M., and M. Merrit (1992). Encrypted key exchange: password-based protocols secure against dictionary attacks. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*. pp. 72–84.

Bellovin, S., and M. Merritt (1993). Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password file compromise. In *The 1st ACM Conference on Computer and Communications Security*. pp. 244–250.

Boyko, V., P. MacKenzie and S. Patel (2000). *Provably Secure Password Authenticated Key Exchange Using Diffie-Hellman.* Eurocrypt 2000.

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

Jablon, D. (1996). Strong password-only authentication key exchange. *ACM Computer Communication Review*, **26**(5), 5–26.

Kwon, T., and J. Song (1999). Secure agreement scheme for g^{xy} via password authentication. *Electronics Letters*, **35**(11), 892–893.

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

Tseng, Y.-M., J.-K. Jan and H.-Y. Chien (2001). On the security of methods for protecting password transmission. *Informatica*, **12**(3), 469–476.

Wu, T. (1998). Secure remote password protocol. In *Internet Society Symposium on Network and Distributed System Security*.

B.-T. Hsieh was born in Taipei, Taiwan, in 1976. He received his BS and MS degrees in information management from Chaoyang University of Technology in 2000. Now, he is currently pursuing his Ph. D. degree in Institute of Information Engineering, National Cheng Kung University, Tainan, Taiwan.

H.-M. Sun received his BS degree in applied mathematics from National Chung–Hsing University in 1988, his MS degree in applied mathematics from National Cheng Kung University in 1990, and his Ph. D. degree in computer science and information engineering from National Chiao–Tung University in 1995, respectively. He was an associate professor with the Department of Information Management, Chaoyang University of Technology from 1995 to 1999, and the Department of Computer Science and Information Engineering, National Cheng Kung University from 1999 to 2002. Currently he is an associate professor with the Department of Computer Science, National Tsing Hua University. He has published over seventy papers. He was the program chair of 2001 National Information Security Conference and the program committee member of 1997 Information Security Conference, 2000 Workshop on Internet and Distributed Systems, Workshop on the 21 st Century Digital Life and Internet Technologies, 1998 and 1999 National Conference on Information Security. His research interests include cryptography, information theory, network security, image compression.

T. Hwang was born in Tainan, Taiwan ROC in March, 1958. He received his undergraduate education from National Cheng Kung University, Taiwan ROC in 1980 and the MS and PhD degrees in Computer Science from the University of Southwestern Louisiana USA in 1988. He is presently a professor of the Department of Information Engineering, National Cheng Kung University, Taiwan ROC. His research interests include cryptology, network security and coding theory. Dr. Hwang is a member of IEEE and also a member of International Association for Cryptographic Research.

Apie kai kurių slaptažodžių tapatumo nustatymo protokolų saugumą

Bin-Tsan HSIEH, Hung-Min SUN, Tzonelih HWANG

Interneto aplinkoje, tokioje kaip UNIX, nutolęs vartotojas prieš pradėdamas bet kokį darbą turi iš serverio gauti prieigos teises. Prieigos teisių gavimo procedūra yra vadinama vartotojo tapatumo nustatymo protokolu. Vartotojo tapatumo nustatymas naudojant vartotojo įsimenamą slaptažodį yra patogus tuo, jog nereikalauja jokių papildomų įtaisų, pvz., mikroprocesorinės kortelės. Vartotojo tapatumo nustatymo protokolas, naudojantis vartotojo prisijungimo vardą ir slaptažodį, iš esmės turėtų atsispirti bandymams atspėti slaptažodį, pavogti tapatybę arba atsisakymo aptarnauti (DoS) atakai. Neseniai Peyravian ir Zunic pasiūlė slaptažodžio perdavimo protokolą ir slaptažodžio pakeitimo protokolą. Vėliau Tseng *et al.* (2001) nurodė, kad Peyravian ir Zunic protokolai negali atsilaikyti prieš bandymus atspėti slaptažodį ir pasiūlė pagerintą protokolą. Nepriklausomai nuo jų, Hwang ir Yeh taip pat pademonstravo Peyravian ir Zunic protokolų saugumo problemas ir pasiūlė pagerintą protokolą. Šiame straipsnyje mes demonstruojame, kad abu Peyravian ir Zunic protokolai ir Tseng *et al.* pagerintas protokolas negali atsispirti prieš bandymus pavogti tapatybę. Be to, mes parodome, kad Peyravian ir Zunic, Tseng *et al.*, Hwang ir Yeh protokolai negali atsispirti prieš atsisakymo aptarnauti ataką.