Cryptanalysis of Two User Identification Schemes with Key Distribution Preserving Anonymity

Eun-Jun Yoon and Kee-Young Yoo[⋆]

Department of Computer Engineering, Kyungpook National University,
Daegu 702-701, South Korea
Tel.: +82-53-950-5553; Fax: +82-53-957-4846
ejyoon@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. In 2004, Wu-Hsu proposed an efficient identification scheme preserving anonymity. However, Yang et al. showed that Wu-Hsu's scheme has a serious weakness, by which the service provider can learn the secret token of the user who requests services. To overcome this limitation, they further proposed a scheme to attain the same set of objectives as the previous works. Nevertheless, the two schemes still have other serious weaknesses. Accordingly, the current paper demonstrates the vulnerability of the two schemes. Furthermore, we present a method to avoid attack.

Keywords: Cryptography, Password, Key establishment, Forward Secrecy.

1 Introduction

In distributed computing environments, it is necessary to maintain user anonymity. That is, only the service provider can identify the user, while no other entity can determine any information concerning the user's identity. In 2000, Lee and Chang [1] proposed a user identification scheme based on the security of the factoring problem [2][3] and the one-way hash function [3][4]. Their scheme has the following advantages: (1) Users can request services without revealing their identities to the public; (2) Each user needs to maintain only one secret; (3) It is not necessary for service providers to record the password files for the users; (4) No master key updating is needed if a new service provider is added into the system.

However, in 2004, Wu-Hsu (WH) [5] showed that Lee-Chang's user identification scheme is insecure under two attacks. First, when a user requests service from a service provider, since only one-way authentication of the user is implemented, an attacker can impersonate the service provider; second, if an expired session key is disclosed, an attacker can break the user anonymity of the corresponding past session. Then they proposed a more efficient identification scheme

[⋆] Corresponding author.

S. Qing et al. (Eds.): ICICS 2005, LNCS 3783, pp. 315–322, 2005.

preserving the same merits [5]. The WH scheme not only effectively eliminates the security leaks of the Lee-Chang scheme, it also reduces computational complexities and communication costs.

Recently, Yang et al. (YWBWD) [6] showed that the WH scheme has a serious weakness, by which the service provider can learn the secret token of the user who requests services. To overcome this limitation, they further proposed a scheme to attain the same set of objectives as the previous works.

However, the WH scheme and YWBWD scheme have other serious weaknesses. Accordingly, the current paper demonstrates the vulnerability of two schemes. Using our attacks, we will show that a malicious user (including the service provider) can easily obtain a specific legal user's secret token and impersonate this specific user to request a service from the service provider and gain access privilege. Additionally, we will show that a malicious user (including the legal user) can easily get the service provider's secret token and impersonate this service provider to exchange a common session key with a legal user. Furthermore, we present an improvement to repair the security flaws of the two schemes.

This paper is organized as follows: In Section 2, we briefly review the WH scheme and YWBWD scheme. Section 3 shows the security flaws of two schemes. In Section 4, we present an improvement of the two schemes. In Section 5, we analyze the security of our proposed scheme. Finally, our conclusions are presented in Section 6.

2 Literature Review

This section separately reviews the WH scheme [5] and YWBWD scheme [6].

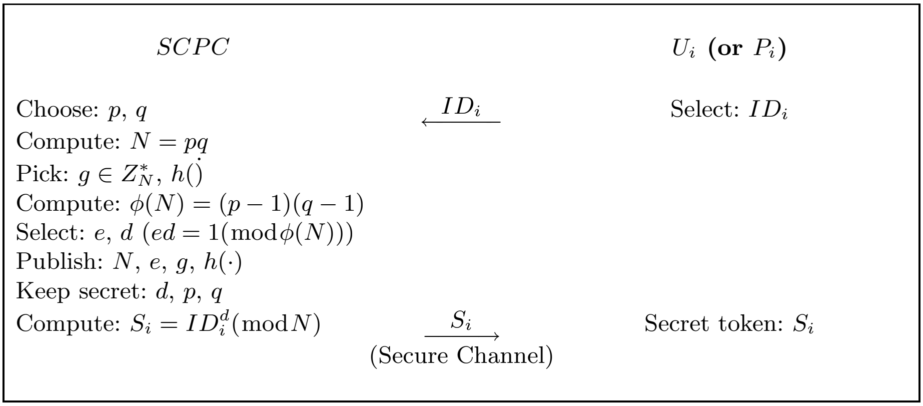
2.1 WH Scheme

The WH scheme is composed of two phases: key generation and anonymous user identification.

Key Generation Phase: The key generation phase of the WH scheme, which is illustrated in Figure 1, is as follows: A smart card producing center ($SCPC$) first chooses two large primes p and q, computes $N = pq$, picks an element $g \in Z_N^*$ and a hash function $h(\cdot)$, and selects e and d such that $ed = 1(\mathrm{mod}\phi(N))$, where $\phi(N)(= (p-1)(q-1))$ is the Euler totient function. N, e, g and $h(\cdot)$ are published and d, p, and q are kept secret by $SCPC$. Then, $SCPC$ sends each user U_i (or service provider P_i) a secret token S_i with a secure channel, where $S_i = ID_i^d(\mathrm{mod}N)$ and ID_i is the identity of U_i (or P_i).

Anonymous User Identification Phase: The anonymous user identification phase of WH scheme, as illustrated in Figure 2, is as follows:

(1) U_i first submits a service request to P_j to request a service from the service provider P_j.



SCPC		U_i **(or** P_i**)**
Choose: p, q	$\xleftarrow{ID_i}$	Select: ID_i
Compute: $N = pq$		
Pick: $g \in Z_N^*$, $h(\dot{})$		
Compute: $\phi(N) = (p-1)(q-1)$		
Select: e, d $(ed = 1(\bmod \phi(N)))$		
Publish: N, e, g, $h(\cdot)$		
Keep secret: d, p, q		
Compute: $S_i = ID_i^d(\bmod N)$	$\xrightarrow{S_i}$ (Secure Channel)	Secret token: S_i

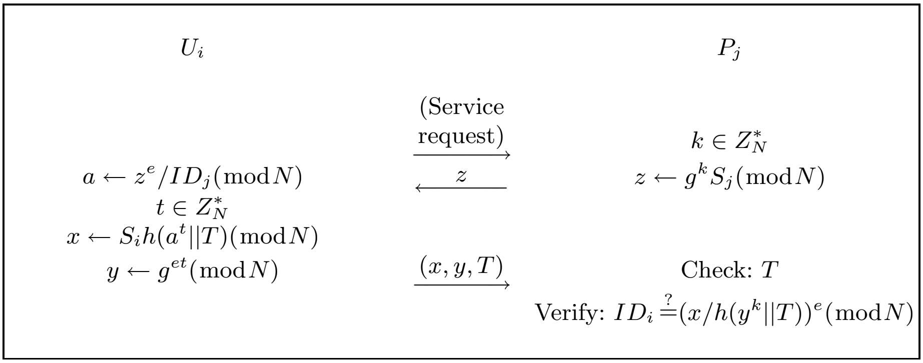
Fig. 1. Key Generation Phase of WH scheme

(2) After receiving the request, P_j chooses a random number k and computes z, where $z = g^k S_j(\bmod N)$. Then, P_j sends z to U_i.

(3) U_i randomly chooses a number t and computes a, x, and y, where $a = z^e/ID_j(\bmod N)$, $x = S_i h(a^t||T)(\bmod N)$, $y = g^{et}(\bmod N)$, and T is the timestamp. Then, U_i sends (x, y, T) to P_j.

(4) Finally, P_j checks T and verifies the equality $ID_i \overset{?}{=} (x/h(y^k||T))^e(\bmod N)$. If it holds for some ID_i existing in the identity list, U_i is accepted as an authorized user and the service request will be granted.

The user and the service provider share common session key as $k_{ij} = a^{tx} = y^{kx} = g^{ektx}(\bmod N)$, which can be used in subsequent communications for confidentiality.



U_i		P_j
	$\xrightarrow{\text{(Service request)}}$	$k \in Z_N^*$
$a \leftarrow z^e/ID_j(\bmod N)$	$\xleftarrow{z}$	$z \leftarrow g^k S_j(\bmod N)$
$t \in Z_N^*$		
$x \leftarrow S_i h(a^t\|\|T)(\bmod N)$		
$y \leftarrow g^{et}(\bmod N)$	$\xrightarrow{(x,y,T)}$	Check: T
		Verify: $ID_i \overset{?}{=} (x/h(y^k\|\|T))^e(\bmod N)$

Fig. 2. Anonymous User Identification Phase of WH scheme

2.2 YWBWD Scheme

To solve the security problem in the WH scheme, Yang et al. proposed an improved version of the WH scheme. The YWBWD scheme is also composed of two phases; key generation and anonymous user identification.

Key Generation Phase: The key generation phase in the YWBWD scheme is similar to that of the WH scheme. The key generation phase of YWBWD scheme, which is illustrated in Figure 3, is as follows: The smart card producing center ($SCPC$) first chooses two large primes p and q, computes $N = pq$, picks an element $g \in Z_N^*$ (which is the generator of both Z_p and Z_q) and a hash function $h(\cdot)$, and selects e and d such that $ed = 1(\mathrm{mod}\phi(N))$, where $\phi(N)(= (p-1)(q-1))$ is the Euler totient function. Note that e must be sufficiently large, e.g., 160 bits. Additionally, $SCPC$ picks a symmetric-key cryptosystem such as DES Schneier, 1996, whose encryption function and decryption function under the private key K are $E_K(\cdot)$ and $D_K(\cdot)$, respectively. N, e, g, and $h(\cdot)$ are published and d, p, and q are kept secret by $SCPC$. Then, $SCPC$ sends each user U_i (or service provider P_i) a secret token S_i with a secure channel, where $S_i = ID_i^d(\mathrm{mod}N)$ and ID_i is the identity of U_i (or P_i).

Anonymous User Identification Phase: The anonymous user identification phase of the YWBWD scheme, which is illustrated in Figure 4, is as follows:

(1) U_i first submits a service request to P_j to request a service from the service provider P_j.
(2) After receiving the request, P_j chooses a random number k and computes z, where $z = g^k S_j^{-1}(\mathrm{mod}N)$. Then, P_j sends z to U_i.
(3) U_i randomly chooses a number t and computes a, K_{ij}, x, s, and y, where $a = z^e ID_j(\mathrm{mod}N)$, $K_{ij} = a^t(\mathrm{mod}N)$, $x = g^{et}(\mathrm{mod}N)$, $s = g^t S_i^{h(x,T)}(\mathrm{mod}N)$,

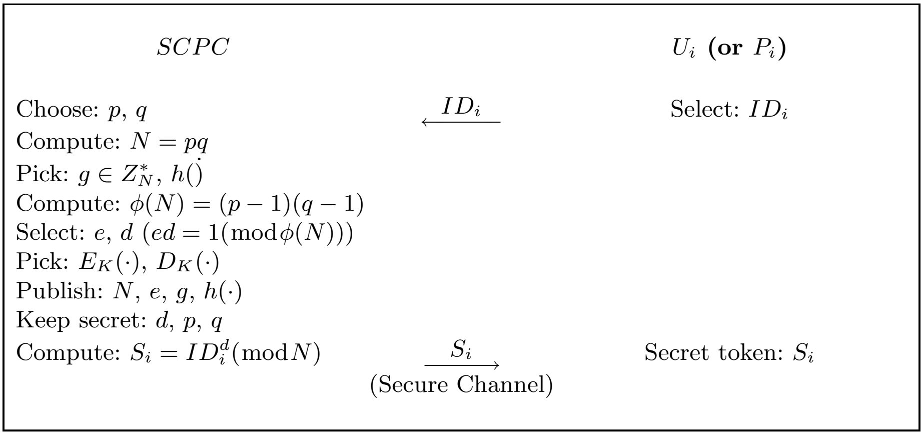

$SCPC$
U_i **(or** P_i**)**
Choose: p, q
$\xleftarrow{ID_i}$
Select: ID_i
Compute: $N = pq$
Pick: $g \in Z_N^*$, $h()$
Compute: $\phi(N) = (p-1)(q-1)$
Select: e, d $(ed = 1(\mathrm{mod}\phi(N)))$
Pick: $E_K(\cdot)$, $D_K(\cdot)$
Publish: N, e, g, $h(\cdot)$
Keep secret: d, p, q
Compute: $S_i = ID_i^d(\mathrm{mod}N)$
$\xrightarrow{S_i}$
(Secure Channel)
Secret token: S_i

Fig. 3. Key Generation Phase of YWBWD scheme

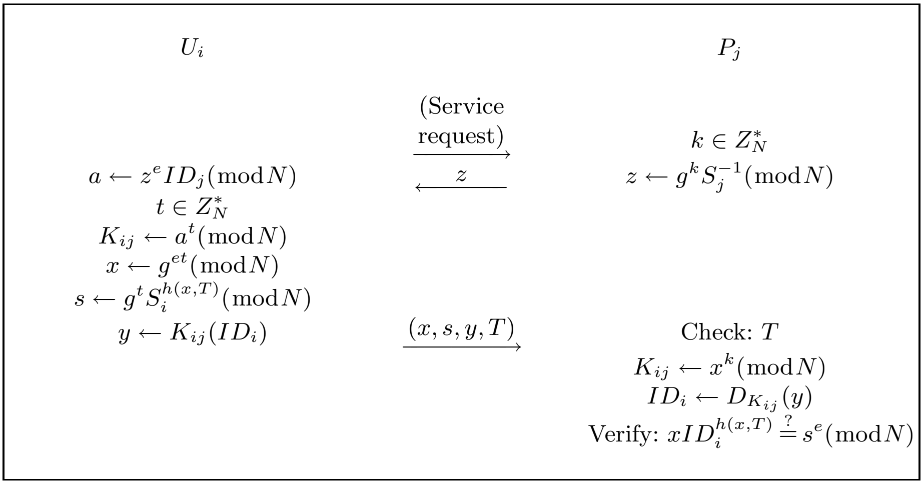

U_i
P_j
(Service request)
$k \in Z_N^*$
$a \leftarrow z^e ID_j (\mathrm{mod} N)$
z
$z \leftarrow g^k S_j^{-1} (\mathrm{mod} N)$
$t \in Z_N^*$
$K_{ij} \leftarrow a^t (\mathrm{mod} N)$
$x \leftarrow g^{et} (\mathrm{mod} N)$
$s \leftarrow g^t S_i^{h(x,T)} (\mathrm{mod} N)$
$y \leftarrow K_{ij}(ID_i)$
(x, s, y, T)
Check: T
$K_{ij} \leftarrow x^k (\mathrm{mod} N)$
$ID_i \leftarrow D_{K_{ij}}(y)$
Verify: $xID_i^{h(x,T)} \overset{?}{=} s^e (\mathrm{mod} N)$

Fig. 4. Anonymous User Identification Phase of YWBWD scheme

$y = K_{ij}(ID_i)$, and T is the timestamp. Then, U_i sends (x, s, y, T) to P_j. Note that K_{ij} is the common session key.

(4) Finally, P_j first checks T. If it is old, P_j aborts the protocol. Otherwise, P_j obtains the common session key $K_{ij} = x^k (\mathrm{mod} N)$. With K_{ij}, P_j proceeds to decrypt y as $ID_i = D_{K_{ij}}(y)$. P_j then checks whether ID_i is on his maintained list. If ID_i is a legitimate user, P_j verifies the equality $xID_i^{h(x,T)} \overset{?}{=} s^e (\mathrm{mod} N)$. If the verification passes, then the service request is granted. Otherwise, the request is rejected.

The user and the service provider share common session key as $k_{ij} = a^t = x^k = g^{ekt} (\mathrm{mod} N)$, which can be used in the subsequent communications for confidentiality.

3 Cryptanalysis of Two Schemes

This section show the security flaws of the WH scheme and YWBWD scheme. In the schemes, an attacker can freely impersonate the users or the service provider. This happens because an attacker can obtain the secret token $S_i(S_j)$ of the user (or the service provider) after successful execution of the key generation phase.

3.1 Attack to U_i

Suppose user U_f is an attacker who knows the legal user U_i's ID_i. Usually, because the legal user's ID_i does not require safety, an attacker can easily get the target user's ID_i by various attack methods, such as stolen-verifier attacks [7] and server data eavesdropping [8]. For example, service provider P_j is always the target of attacker, because numerous users' secrets are stored in their databases.

The user ID table list stored in the service provider P_j can be eavesdropped and then used to impersonate the original user. By using the legal user U_i's ID_i, in the key generation phase, U_f can register with $SCPC$ as follows:

(1) U_f obtains his/her identity ID_f by $ID_f = ID_i^{-1}$ and submits ID_f as registration request to $SCPC$.
(2) $SCPC$ will compute the secret token S_f of U_f by $S_f = ID_f^d = ID_i^{-d} = S_i^{-1}(\text{mod}N)$ and send S_f to U_f with a secure channel.

As a result, U_f can obtain the secret token S_i of the legal user U_i by computing $S_f^{-1} = S_i(\text{mod}N)$. Then, by using the S_i, so obtained, U_f can freely impersonate U_i to request a service from P_j and thus gain access privilege.

3.2 Attack to P_j

Suppose user U_f is an attacker who knows the the service provider P_j's ID_j. In the key generation phase, U_f can register with $SCPC$ as follows:

(1) U_f obtains his/her identity ID_f by $ID_f = ID_j^{-1}$ and submits ID_f as registration request to $SCPC$.
(2) $SCPC$ will compute the secret token S_f of U_f by $S_f = ID_f^d = ID_j^{-d} = S_j^{-1}(\text{mod}N)$ and send S_f to U_f with a secure channel.

As a result, U_f can obtain the secret token S_j of the service provider P_j by computing $S_f^{-1} = S_j(\text{mod}N)$. Then, by using obtained S_j, U_f can impersonate P_j and exchange a common session key with legal user U_i.

3.3 Another Attack

As another attack on two schemes, if a malicious U_i or P_j, who knows his/her S_i or S_j, computes his/her new identity ID_f by $ID_f = ID_iID_j$ and resubmits ID_f as a registration request to $SCPC$. Then, $SCPC$ will compute the secret token S_f of U_f by $S_f = ID_f^d = (ID_iID_j)^d = S_jS_j(\text{mod}N)$ and send S_f to U_f with a secure channel. Consequently, a malicious U_i can obtain the secret token S_i of the legal user U_i or S_j of the service provider P_j by computing $S_i = S_fS_j^{-1}(\text{mod}N)$ or $S_j = S_fS_i^{-1}(\text{mod}N)$, respectively.

4 Improved Scheme

This section presents a modification of the two schemes to correct the security flaws described in Section 3.

The proposed scheme employs the concept of hiding identity to prevent from above attacks. We only modify the key generation phase which issues a "hashed" identity for every legal user. That is, in the key generation phase, the smart card

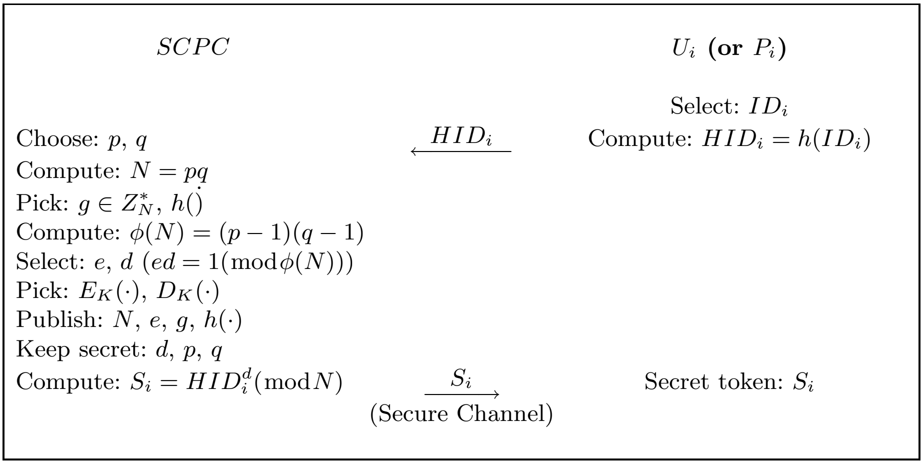

$SCPC$
U_i (or P_i)
Select: ID_i
Choose: p, q
$\xleftarrow{HID_i}$
Compute: $HID_i = h(ID_i)$
Compute: $N = pq$
Pick: $g \in Z_N^*$, $h()$
Compute: $\phi(N) = (p-1)(q-1)$
Select: e, d $(ed = 1(\bmod \phi(N)))$
Pick: $E_K(\cdot)$, $D_K(\cdot)$
Publish: N, e, g, $h(\cdot)$
Keep secret: d, p, q
Compute: $S_i = HID_i^d(\bmod N)$
$\xrightarrow{S_i}$
(Secure Channel)
Secret token: S_i

Fig. 5. Proposed Key Generation Phase

producing center ($SCPC$) sends each user U_i (or service provider P_i) a secret token $S_i = HID_i^d(\bmod N)$ with a secure channel, where $HID_i = h(ID_i)$. The steps of the anonymous user identification phase are retained except that ID_i is replaced by "hashed" identity HID_i, respectively. The proposed key generation phase is illustrated in Figure 5.

5 Security Analysis

This section discusses the enhanced security features. The rest are the same as the original YWBWD scheme as described in the literature [6]. Readers are referred to [6] for completer references.

Definition 1. *One-way hash function assumption [3,4,9]: Let $h(\cdot)$ be an one-way cryptographic hash function, (1) given y, it is computationally intractable to find x such that $y = h(x)$; (2) it is computationally intractable to find $x_1 \neq x_2$ such that $h(x_1) = h(x_2)$.*

Theorem 1. *In the proposed key generation phase, an illegal user cannot get the legal user or service provider's secret token S_i.*

Proof. The attacks on WH scheme and YWBWD scheme works because a malicious user can successfully register a new ID_f via ID_i or ID_j in the key generation phase. In our improved key generation phase, since the format of $HID_f^d = h(ID_i^{-1})^d(\bmod N)$ (or $h(ID_iID_j)^d(\bmod N)$) is not equal to $ID_f^d = ID_i^{-d}(\bmod N)$ (or $S_iS_j(\bmod N)$), a malicious user cannot get the legal user or service provider's secret token S_i. Therefore, the proposed scheme can correct the security flaws described in Section 3.

6 Conclusions

The current paper demonstrated the security flaws of the WH user identification scheme and YWBWD user identification scheme. Using our attacks, we have shown that a malicious user (including a service provider) can easily get a specific legal user's secret token and impersonate this specific user to request a service from the service provider and gain access privilege. Additionally, we have shown that a malicious user (including the legal user) can easily get the service provider's secret token and impersonate this service provider to exchange a common session key with a legal user. For the above attacks, we presented an improvement to repair the security flaws of the two schemes.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments in improving our manuscript. This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

References

1. Lee, W.B., Chang, C.C.: User Identification and Key Distribution Maintaining Anonymity for Distributed Computer Network. Computer Systems Science and Engineering. Vol. 15. No. 4. (2000) 113-116
2. Rivest, R., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signature and Public-key Cryptosystem. Commun ACM. Vol. 21. No. 2. (1978) 120-126
3. Schneier, B.: Applied Cryptography. 2nd ed. John Wiley & Sons. Inc. (1996)
4. Diffie, W., Hellman, M.: New Directions in Cryptography. IEEE Trans Inf Theory. Vol. 22. No. 6. (1976) 644-654
5. Wu, T.S., Hsu, C.L.: Efficient User Identification Scheme with Key Distribution Preserving Anonymity for Distributed Computer Networks. Computer & Security. Vol. 23. No. 2. (2004) 120-125
6. Yang, Y.J., Wang, S.H., Bao, F., Wang, J., Deng, R.H.: New Efficient User Identification and Key Distribution Scheme Providing Enhanced Security. Computer & Security. Vol. 23. No. 8. (2004) 697-704
7. Lin, C.L., Hwang, T.: A Password Authentication Scheme with Secure Password Updating. Computers & Security. Vol. 22. No. 1. (2003) 68-72
8. Yang. C.C., Chang. T.Y., Li, J.W.: Security Enhancement for Protecting Password Transmission. IEICE Transactions on Communications. Vol. E86-B. No. 7. (July 2003) 2178-2181
9. Menezes, A.J., Oorschot, P.C., Vanstone, S.A.: Handbook of Applied Cryptograph. CRC Press. New York. (1997)