Abstract

It is proved that the LHL-key authentication scheme, proposed by Lee, Hwang and Li [Appl. Math. Comput. 139 (2003) 343], is insecure. The user’s private key can be obtained easily from the user’s public key certificate, hence compromising all the encrypted communications. In addition, the certificate validation process proposed in the same work is not a suitable one, as any certificate (valid or not) satisfies the verification equation. A slight modification is pointed out to overcome this severe weakness. © 2003 Elsevier Inc. All rights reserved.

Keywords: Cryptanalysis; Key authentication scheme; Public-key cryptography

1. Introduction

The LHL-key authentication scheme (LHL-scheme) was proposed by Lee, Hwang and Li [1] to resolve the security problems in two previous key authentication schemes: the HY-scheme [2] and the ZLYH-scheme [3]. All of these key authentication schemes are based on the password table hosted in the server allowing the users to log into the system. The ZLYH-scheme is an improvement of HY-scheme to prevent the guessing attack [4], while LHL-scheme is a modification of ZLYH-scheme to provide non-repudiation service.

In the next sections, the LHL-scheme is introduced and several aspects on its security are analysed, proving the existence of severe weaknesses, concluding
that LHL-scheme is not secure and that non-repudiation (as claimed in [1]) is not provided.

2. The LHL-scheme

Every user has a password PWD to log into the system, and a pair of keys, the private key Prv and the public key Pub, to employ an asymmetric cryptosystem. The public key is computed as

$$\text{Pub} = g^{\text{Prv}} \mod p$$  \hspace{1cm} (1)

where $p$ is a large prime, and $g$ is a generator in $\mathbb{Z}_p^*$. The prime $p$, the generator $g$ and a one-way function $f$ are public parameters, where $f$ is defined as

$$f(x) = g^x \mod p$$  \hspace{1cm} (2)

In the registration phase, the user’s public key certificate is generated by the user from his password and private key. Each user chooses a random number $r$ in $\mathbb{Z}_p^*$ such that $\gcd((\text{PWD} + r), \text{Prv}) = 1$, allowing to compute the integers $a, b$ satisfying the equation

$$a(\text{PWD} + r) + b\text{Prv} = 1$$  \hspace{1cm} (3)

The user sends $f(\text{PWD} + r), R = g^r \mod p, a$ and $b$ to the server secretly. The server verifies if $f(\text{PWD} + r) = f(\text{PWD})R \mod p$ and $f(\text{PWD} + r)^a\text{Pub}^b = g \mod p$. Then, $f(\text{PWD} + r), a$ and $b$ are stored in the public password table hosted in the server. Note that it is assumed in [1] that this table cannot be modified. Moreover, the certificate $C$ of the user’s public key is generated as follows

$$C = \frac{(\text{PWD} + r)}{f(\text{PWD} + r) + \text{Prv}} \mod (p - 1)$$  \hspace{1cm} (4)

The certificate $C$ and the public key Pub are opened to public in the network.

In the authentication phase, when a user wants to communicate to another user, the sender obtains the receiver parameters $C$ and Pub from the network and $f(\text{PWD} + r), a$ and $b$ from the public password table. Next, the sender checks $C$ by means of the following equation

$$f(C) = f(\text{PWD} + r)^a\text{Pub}^b \mod p$$  \hspace{1cm} (5)

If this equation holds, the sender accepts the public key Pub to encrypt the messages.
3. Cryptanalysis of the LHL-scheme

As it is described in [1], the parameters \( f(PWD + r), a, b, \text{Pub} \) and \( C \) are public, and any user (malicious or not) can obtain them from the public password table and the network. These known parameters and Eqs. (3) and (4) allow the intruder to calculate easily the private key of any user. The procedure is as follows. Since the unknown parameter \( (PWD + r) \) can be expressed as

\[
(PWD + r) = \frac{1 - bPrv}{a}
\]  

the certificate equation (4) can be rewritten as

\[
C = \frac{1 - bPrv}{af(PWD + r) + aPrv} \mod (p - 1)
\]

Hence, we have

\[
Prv = \frac{1 - f(PWD + r)aC}{aC + b} \mod (p - 1)
\]

Let us see an example. We consider \( p = 46549, g = 1257, \text{Prv} = 6532, \text{Pub} = g^{\text{Prv}} \mod p = 14972 \) and \( (PWD + r) = 7845 \). Since \( \gcd(PWD + r, \text{Prv}) = 1 \), \( a(PWD + r) + b\text{Prv} = 1 \) is satisfied for \( a = -2771 \) and \( b = 3328 \). The certificate \( C \) is calculated from the value of \( f(PWD + r) = 12619 \), producing \( C = 11271 \). If we apply Eq. (8), we have \( \text{Prv}' = 6532 = \text{Prv} \).

On the other hand, Eq. (5) can not be considered as a certificate validation process, since any value of \( C \) in \( \mathbb{Z}_p \) is considered a valid certificate for the correct values of \( f(PWD + r), \text{Pub}, a \) and \( b \). In other words, the certificate \( C \) computed from Eq. (4) has no effect on the user's public key authentication, as it is shown in Eq. (9), because the verification result is independent to the value of \( C \).

\[
f(C) = f(PWD + r)^a\text{Pub}^b \mod p = g^{C(PWD + r)a + b\text{Pub}} \mod p = g^C \mod p
\]

In addition, when \( \gcd(f(PWD + r) + \text{Prv}, p - 1) > 1 \) the certificate can not be generated by Eq. (4), since the inverse of \( f(PWD + r) + \text{Prv} \mod (p - 1) \) does not exist.

4. Modifications to the LHL-scheme

In order to overcome the security weaknesses, the following modifications to the LHL-scheme are proposed.
(a) Not to use the certificate parameter $C$, in order to avoid the easy computation of Prv. In this way, the security analysis results in [1] can be applied to this case.

(b) To apply Eq. (10) instead of Eq. (5) as user's public key verification (authentication) process. In this way, the content of the inalterable public password table can be considered as the public key certificates. Note that Eq. (10) is the same equation used by the server to check the validity of parameters in the user registration phase.

$$f(PWD + r)^aPub^b = gmod p$$

5. Conclusions

The LHL-scheme proposed in [1] is not secure as the intruder can obtain the private key of the users. The modifications proposed in this letter prevent from this problem, maintaining the features claiming in [1], that is, the guessing attack prevention and the user's key non-repudiation.

Acknowledgements

This work have been supported by MICYT-Spain under grant TIC2001-0586, “Gestión del Acceso Seguro a redes abiertas de recursos distribuidos”.

References


