2011 International Conference on Network and Electronics Engineering
IPCSIT vol.11 (2011) © (2011) IACSIT Press, Singapore

Security Improvement against Malicious Servers in dPEKS Scheme

[1]Bing-Jian Wang, [1]Tzung-Her Chen, [2*]Fuh-Gwo Jeng

[1]Department of Computer Science and Information Engineering
[2]Department of Applied Mathematics
National Chiayi University, Chiayi City, Taiwan 60004, R.O.C.
*E-mail: fgjeng@mail.ncyu.edu.tw

Abstract. While the original public-key encryption with keyword search scheme (PEKS) has been pointed out to be insecure against off-line keyword-guessing attacks, some public-key encryption with designated tester schemes (dPEKS) proposed recently also encounter the same attacks. Rhee et al. proposed a dPEKS which is intended to prevent the off-line keyword-guessing attacks. However, we find that the off-line keyword-guessing attacks are still working in the test phase when some malicious servers exist. Hence, we add a random parameter into the test phase of Rhee et al.'s scheme to get a more secure and improved dPEKS scheme so as to prevent from keyword-guessing attacks and to benefit the advantages of dPEKS as well.

Keywords: searchable encryption, designated tester, data security

1. Introduction

To protect the confidentiality of sensitive data in clouding-computing environments, a reliable encryption technology is used to encrypt the sensitive data stored in the server. For a user issuing a keyword searching on the encrypted data, the server unavoidably faces the security problem of how to process the search without revealing any sensitive information. Especially, the server maintaining the database of encrypted data is not trusty.

The public-key encryption with keyword search scheme (PEKS) is first proposed by Boneh et al. (2004) [1]. Based on Boneh et al.'s scheme, Hwang and Lee (2007) [2] proposed another PEKS scheme for multi-receiver. The concept of proxy re-encryption is applied in keyword search by Shao et al. (2010) [3] and by Yau and Phan (2010) [4] as well. Recently, a conjunctive subset keywords search is proposed by Zhang et al. (2011) [5]. However, Baek et al. (2006) [6] pointed out that an outside attacker in the PEKS scheme could perform the test process by collecting the transmitted ciphertexts and trapdoors. Thus the attacker could further construct the relationship between encrypted data and the given trapdoors of known keywords. Therefore, Baek et al. [6] proposed their public-key encryption scheme with designated tester (dPEKS) to solve the problem. In the same year, Byun et al. [7] pointed out that the design of trapdoors in PEKS scheme was insecure against off-line keyword-guessing attacks. Because an attacker can choose a keyword to test whether the captured trapdoor includes the guessed keyword with the receiver's public key and bilinear map operation, the interested keyword of the receiver is revealed. Unfortunately, although Baek et al.'s dPEKS scheme [6] achieves tester designating, the trapdoor's structure is the same with that in PEKS's. Inheritably, their scheme cannot prevent off-line keyword-guessing attacks. Therefore based on Baek et al.'s dPEKS scheme [6], Rhee et al. (2010) [8] enhanced the trapdoor security so as to prevent from off-line keyword-guessing attacks.

Rhee et al. [8] claimed that their dPEKS scheme with a new trapdoor function was secure against keyword-guessing attacks. Yet, we would like to point out their trapdoor design was still on the risk of

keyword-guessing attacks especially by malicious servers in this paper. In the further, we will present an improved dPEKS scheme to avoid this drawback.

2. Cryptanalysis and improvement of Rhee et al.'s dPEKS scheme)

In this section, we would like to review the Rhee et al.'s dPEKS scheme [8] first and then try to point out their security problems.

2.1. Rhee et al.'s dPEKS scheme

1) Global setup

Determine two cyclic groups G_1 and G_2 with prime order p, and their admissible bilinear paring function $\hat{e}: G_1 \times G_1 \to G_2$. Define three hash functions as $H: \{0,1\}^* \to G_1$, $H_1: \{0,1\}^* \to G_1$ and $H_2: G_2 \to \{0,1\}^{\lambda}$, where λ is a security parameter. Choose a random generator $P \in G_1$.

2) Key generation for server and receiver

The server (resp. receiver) generates his private key by randomly choosing $sk_S = x \in \mathbb{Z}_p$ (resp. $sk_R = y \in \mathbb{Z}_p$) and the corresponding public key by computing $pk_S = P^x$ (resp. $pk_R = P^y$).

3) dPEKS: $(pk_R, pk_S, W) \to (U, V)$

The sender adopts receiver's and server's public keys, pk_R and pk_S, to compute the dPEKS ciphertext by $(U,V) = (pk_R{}^r, H_2(\hat{e}(pk_S, H_1(W)^r))$, where W is keyword, and $r \in \mathbb{Z}_p$ is randomly chosen. Then the ciphertext (U,V) is sent to server for receiver's search later.

4) dTrapdoor: $(pk_S, sk_R, W') \to T_{W'}$

When the receiver intends to process the search for keyword W', he has to generate a trapdoor for the keyword by computing $T_{W'} = (T_1, T_2) = (P^{r'}, H_1(W')^{1/y} \cdot H(pk_S{}^{r'}))$, where $r' \in \mathbb{Z}_p$ is randomly chosen. Then the receiver sends $T_{W'}$ to the server for searching process.

5) dTest: $((U,V), sk_S, T_{W'}) \to Boolean$

After the server receives the trapdoor $T_{W'}$ from the receiver, the server is able to test whether the keyword W' exists in some ciphertext (U,V) or not. First, the server computes $T_3 = T_2 / H(T_1^{sk_S}))$. Second, the server checks if $H_2(\hat{e}(U, T_3^{sk_S}))$ is equal to V. If yes, the server sends the search result to the receiver.

2.2. Security problem

In the test phase of Rhee et al.'s dPEKS scheme, the server can compute $T_3 = T_2 / H(T_1^{sk_S})) = H_1(W')^{1/y}$. Accordingly, we found that a malicious server can forwardly compute $\hat{e}(pk_R, T_3) = \hat{e}(P^y, H_1(W')^{1/y}) = \hat{e}(P, H_1(W'))$. Then the malicious server can perform a keyword-guessing attack with $\hat{e}(P, H_1(W'))$ to guess which keyword the receiver is interested in.

2.3. Improvement of Rhee et al.'s dPEKS scheme

To prevent the risk of keyword-guessing attacks from a malicious server, a random number u is introduced into the trapdoor function computed by the receiver; that is, $T_{W'} = (T_1, T_2) = (P^{r'}, H_1(W')^{u/y} \cdot H(pk_S{}^{r'}))$, where $u \in \mathbb{Z}_p$ is randomly chosen. The rest phases of the original dPEKS are kept the same.

3. Concluding remarks

Since our improvement is on the trapdoor phase only, the security analysis will focus on the improvement itself.

Obviously, a malicious server can compute $T_3 = T_2 / H(T_1^{sk_S})) = H_1(W')^{u/y}$ and $\hat{e}(pk_R, T_3) = \hat{e}(P^y, H_1(W')^{u/y}) = \hat{e}(P, H_1(W'))^u$. However, without the random number u chosen by the receiver, the server cannot perform the keyword-guessing attack only with $\hat{e}(P, H_1(W'))^u$ in hand.

In this paper, we have pointed out the Rhee et al.'s dPEKS scheme [8] with the weakness against off-line keyword-guessing attacks by malicious servers. Hence, an improved dPEKS scheme is presented to prevent the attacks from malicious servers.

Acknowledgement

This work was partially supported by National Science Council under Grant No. NSC 99-2221-E-415-017 and NSC 99-2221-E-415-008.

4. References

[1] D. Boneh, G.D. Crescenzo, R. Ostrovsky, G. Persiano, “Public key encryption with keyword search,” Proceedings of EUROCRYPT’04. LNCS, Vol. 3027, 2004, pp. 506–522.

[2] Y.H. Hwang, P.J. Lee, “Public key encryption with conjunctive keyword search and its extension to a multi-user system,” Proceedings of Pairing 2007, LNCS, Vol. 4575, 2007, pp. 2–22.

[3] J. Shao, Z.F. Cao, X.H. Liang, H. Lin, “Proxy re-encryption with keyword search,” Information Sciences, Vol. 180, Issue 13, 1 July 2010, pp. 2576-2587.

[4] W.C. Yau, R.C.-W. Phan, S.H. Heng, B.M. Goi, “Proxy re-encryption with keyword search: new definitions and algorithms,” Communications in Computer and Information Science, Vol. 122, 2010, pp. 149-160.

[5] B. Zhang, F. Zhang, “An efficient public key encryption with conjunctive-subset keywords search,” Journal of Network and Computer Applications, Vol. 34, Issue 1, January 2011, pp. 262-267.

[6] J. Baek, R. Safavi-Naini, W. Susilo, “Public key encryption with keyword search revisited,” Proceedings of ACIS’06, 2006.

[7] J.W. Byun, H.S. Rhee, H.A. Park, D.H. Lee, “Off -line keyword guessing attacks on recent keyword search schemes over encrypted data,” Proceedings of SDM’06. LNCS, Vol. 4165, 2006, pp. 75–83.

[8] H.S. Rhee, J.H. Park, W. Susilo, D.H. Lee, “Trapdoor security in a searchable public-key encryption scheme with a designated tester,” Journal of Systems and Software, Vol. 83, Issue 5, May 2010, pp. 763-771.