Lu-Cao 패스워드기반 키 교환 프로토콜의 안전성 분석

Cryptanalysis on Lu-Cao's Key Exchange Protocol

윤택영, 조성민, 박영호*
(Taek-Young Youn, Sung Min Cho and Young-Ho Park)

Abstract : Recently, Lu and Cao proposed a password-authenticated key exchange protocol in the three party setting, and the authors claimed that their protocol works within three rounds. In this paper, we analyze the protocol and show the protocol cannot work within three rounds. We also find two security flaws in the protocol. The protocol is vulnerable to an undetectable password guessing attack and an off-line password guessing attack.

Keywords: Cryptanalysis, Password-Authenticated Key Exchange, Three-Party Setting.

I. Introduction

Password-authenticated key exchange (PAKE) protocol allows two communicating parties to share a session key over an insecure channel. In 1992, encrypted key exchange (EKE) protocol, which is the first version of PAKE, has been proposed by Bellovin and Merritt [2]. In PAKE, two communicating entities identify the communicating partner using a shared password. Hence they should share a password prior to establish a common session key under the protocol. In this case, each entity may store many passwords to communicate with several entities. Then the number of passwords that stored by an entity corresponds to the number of clients who are expected to establish a session key with the entity. It is inconvenient to remember all the passwords since the number of human-memorable string is limited. To solve the problem, three-party password-authenticated key exchange protocol (3PAKE) has been proposed which allows two clients establish a session key without share a common password. In the setting, each client shares a password with a trusted server, and the server helps two clients to establish a session key. The main advantage of this solution is that it provides each user with the capability of communicating securely with other users in the system while only requiring it to remember a single password. Though the server should participate in the procedure for establishing a session key between two clients, each client can share a common session key using different passwords. Until now, several papers [1,3,4,5,8,9] have considered password-based key exchanges in the three-party setting.

For 3PAKE, round efficiency is important when session keys are exchanged frequently, because the server should participate in the execution of a protocol. Especially, for mobile applications, to reduce the number of round is very important, because the mobile networks have limited bandwidth. Until now, many protocols have been proposed to reduce the number of rounds. Recently, Lu and Cao proposed a round efficient protocol [6] which requires only three rounds.

In this paper, we analyze the protocol in [6]. Firstly, we show that Lu and Cao's protocol can not work within three rounds. Moreover, we also show that the protocol has two weaknesses. The protocol is not secure against an undetectable on-line password guessing attack and off-line password guessing attack.

II. Review of Lu and Cao's

In this section, we review the protocol in [6]. System parameters are defined as follows. Let p,q be prime numbers such that p=2q+1, and G be a cyclic group of order q. Let g, s and t be distinct generators of G. Let H:$\{0,1\}^* \rightarrow$ Zp and H':$\{0,1\}^* \rightarrow \{0,1\}^k$ be two distinct hash functions that map a string of arbitrary length to an element of Zp and k-bit string, respectively. In [6], two hash functions are not precisely described, and so we use usual settings. Note that, in [6], Lu and Cao argued that their protocol works within three rounds, but it requires actually five rounds. Hence, we re-arrange the execution of the protocol as follows.

Round 1: A client A chooses a random x in Zq and computes $X=$ g^xs^{pw1}. Note that, the identity and password of A are id_A and pw1, respectively. Similarly, we define that the identity and password of B as id_B and pw2. A initiates a protocol by sending $id_A||X$ to B.

Round 2: When B receives $id_A||X$, it chooses y in Zq and computes $Y=g^yt^{pw2}$. Then B sends $id_A||X||id_B||Y$ to S.

* 책임저자(Corresponding Author)

논문접수 : 20xx. x. x., 채택확정 : 200x. x. xx.

윤택영, 조성민 : 고려대학교 정보경영공학전문대학원
박영호 : 세종 사이버대학교
(taekyoung@cist.korea.ac.kr, muji0828@naver.com, young-ho@cybersejong.ac.kr)

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음" (IITA-2008-(C1090-0801-0025))

Round 3: S computes g^x=X/s$^\wedge$^{pw1} and g^y=Y/t^{pw2}. S chooses a random z in Zq, and computes $(g^x)^z$ and $(g^y)^z$. Let id_S be the identity of the server S. Then, S sends X'||Y' to B, where
X'=g^{yz}H(id_A, id_S,g^x) ^{pw1} and
Y'=g^{xz}H(id_B, id_S,g^y) ^{pw2}.

Round 4: B computes g^{xz} = Y'/H(id_B, id_S,g^y) ^{pw2}, g^{xyz}=$(g^{xz})^y$ and alpha=H(id_A, id_B,g^{xyz}), and sends X'||alpha to A.

Round 5: A computes g_{yz}=X'/H(id_A, id_S,g^x) ^{pw1} and g^{xyz}=$(g^{yz})^x$. He checks whether alpha =H(id_A, id_B,g^{xyz}) holds or not. If it does not hold, A rejects the protocol. Else, A computes beta=H(id_B, id_A,g^{xyz}) and sends it to B.

Verification Phase: B checks whether beta = H(id_B, id_A,g^{xyz}) holds or not. If it does not hold, B rejects the protocol. If the protocol is not rejected until the final round, A and B compute the session key as sk=H'(id_A, id^B,g^{xyz}).

III. Count the Number of Round

In this section, we show that the protocol in [6] cannot work within three rounds. To count the number of round, we adopt two notions, a step and a round, which are defined in [7]. One step is the event that one party sends communicating messages to a single party at one time. A round is defined as a set of all independent steps that can be processed in parallel. Note that, Lu and Cao also use the same definition of round. Though they did not mention about the notions, they compare their protocol with the protocol proposed in [7] in terms of round efficiency under the same measure.

Under the definition of round, we can see that two or more parties can send their communicating messages simultaneously in a round only if each communicating message can be generated using a set of previously obtained data. Under the observation, we explain the reason why the protocol in [6] requires actually five rounds rather than three rounds.

In [6], Lu and Cao misuse the concept of round. By definition, a round means a set of all independent steps that can be executed in parallel. However, some processes are tied up where they can not executed simultaneously. In [6], Lu and Cao claimed that Round 1 and Round 2 are executed in one round. However, B can not send id_A||X||id_B||Y before B receives id_A||X from A in Round 1. Moreover, they claim that Round 3 and Round 4 are executed in one round, but, B can send Y'||alpha to A only if B receives X'||Y' from S. Consequently, their protocol requires five rounds.

IV. Cryptanalysis of Lu and Cao's Protocol

1. Undetectable On-Line Password Guessing Attack

Let E be an adversary whose goal is to find A's password, and we assume that E has it's own valid password pw_E. E executes the following steps to mound an undetectable on-line password guessing attack.

Step 1: E guesses the password of A pw_A', computes X=$g^x s^{pwA'}$ and Y=$g^y t^{pwE}$ for x, y in Zq, and sends id_A||X||id_E||Y to S. Then, S computes X' =g^{yz}H(id_A, id_S,X/s^{pwA}) ^{pwA} and Y' = $(X/s^{pwA})^z$H(id_E, id_S,g^y) ^{pwE} for z in Zq, and sends them to E.

Step 2: For given X'||Y', E tests if the following holds: (X'/H(id_A, id_S,g^x) $^{pwA'})^x$=(Y'/H(id_E, id_S,g^y) $^{pwE})^y$. The equation holds only if the guessed password is correct, so E can check whether a guessed password is correct or not.

When E tries to test if a guessed password is correct or not, A can not recognize that someone tries to find its password, since S sends all communicating messages only to E. Moreover, S also can not recognize E's attack, since the server does not explicitly verify two clients.

2. Off-Line Password Guessing Attack

If a password of A is odd, an adversary E can mount an off-line guessing attack. Since H is not precisely described in [6], we define it as in Section 2.

For a message m, H(m) is an element of G or Zp/G = {x | x in Zp but not in G}. It is easy to see that x^q = 1 mod p for x in G, and x^q $\neq$ 1 mod p for x in Zp/G. E collects X and X' from a valid execution of the protocol. Note that, X'^q $\neq$ 1 mod p if and only if H(id_A, id_S,X/s^{pwA})^q $\neq$ 1 mod p and pw_A is odd. If X'^q $\neq$ 1 mod p, E guesses an odd password pw_A', and computes tau = H(id_A, id_S,X/$s^{pwA'}$)^q mod p. Note that, the condition X'^q $\neq$ 1 mod p holds only if the password is odd. Hence, in this case, all even passwords can be excluded from the set of probable passwords. If tau = 1, E discards pw_A' from probable password space. The adversary tests for all candidate passwords. Note that, Pr[x in G]=Pr[x in Zp/G]= 1/2, since |G|=|Zp/G|. Hence, E can exclude 50% passwords from the set of candidate passwords which holds tau = 1. In this context, we want to emphasis that a hash function should be designed carefully. If the hash function is designed as H:{0,1}*→ G, the protocol can be secure against our attack.

V. Conclusion

In this paper, we analyzed the simple three-party key exchange protocol, proposed by Lu and Cao, and showed that it is insecure and it can not provide key exchange within three rounds.

참고문헌

[1] M. Abdalla, P.-A Fouque, and D. Pointcheval, ``Password-Based Authenticated Key Exchange in the Three-Party Setting", Proc. of PKC'05, Springer-Verlag, LNCS 3386, pp. 65-84, 2005.

[2] S. M. Bellovin, M. Merritt, ``Encrypted key exchange: password-based protocols secure against dictinary attacks", Proc. of 1992 IEEE Symposium on Research in Security and Privacy, pp. 72-84, 1992.

[3] J. W. Byun, I. R. Jeong, D. H. Lee, and C.-S. Park, ``Password-authenticated key exchange between clients with different passwords", Proc. of ICICS'02, Springer-Verlag, LNCS 2513, pp. 134-146, 2002.

[4] C.-C. Chang, Y.-F. Chang, ``A novel three-party encrypted key exchange protocol", Computer Standards and Interfaces, 26(5), pp. 471-476, 2004.

[5] J. O. Kwon, K. Sakurai, and D. H. Lee, ``Efficient Password-Authenticated Key Exchange for Three-Party Secure Against Undetectable On-Line Dictionary Attacks", Proc. of ICCS'06, Part 1, Springer-Verlag, LNCS 3991, pp. 977-980, 2006.

[6] R. Lu, and Z. Cao, ``Simple three-party key exchange protocol," Computers & Security, Volume 26, pp. 94-97, 2007.

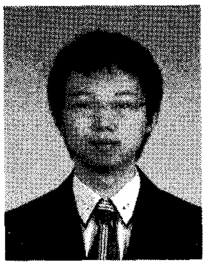
[7] T.-F. Lee, T. Hwang, C.-L. Lin, ``Enhanced three-party encrypted key exchange without server public keys.", Computers & Security, Volume 23, pp. 571-577, 2004.

[8] C.-L. Lin, H.-M. Sun, and T. Hwang, ``Three-party encrypted key exchange: Attacks and a solution", ACM SIGOPS Operating Systems Review, 34(4), pp. 12-20, Oct. 2000.

[9] C.-L. Lin, H.-M. Sun, M. Steiner, T. Hwang, ``Three-party encrypted key exchange without server public keys", IEEE Communication Letters, 5(12), pp. 497-499, 2001.

윤 택 영

2003년: 고려대학교 수학과 이학학사
2005년: 고려대학교 정보보호대학원 정보보호학과 공학석사
2005년 3월~현재: 고려대학교 정보보호대학원 정보보호학과 박사과정.
<관심분야> 암호 이론, 정보보호 이론, 암호 프로토콜, 부채널 공격



조 성 민

2008년: 광운대학교 수학과 이학학사
2008년 3월~현재: 고려대학교 정보보호대학원 정보보호학과 석사과정
<관심분야> 암호 이론, 정보보호 이론, 암호 프로토콜, 부채널 공격

박 영 호ss

1990 년: 고려대학교 수학과 이학사
1993 년: 고려대학교 수학과 이학석사
1997 년: 고려대학교 수학과 이학박사
2002 년~현재: 세종 사이버 대학교 조교수
<관심분야> 정수론, 공개키 암호, 암호 프로토콜, 부채널 공격

※ 저자의 순서는 상좌(1번째), 상우(2번째), 하좌 (3번째)의 순으로 한다.