Cryptanalysis of Sui *et al.*'s Second ID-based Key Issuing Protocol without Key Escrow★

Hyunjue Kim, Seungjoo Kim, and Dongho Won

Information security Group,
School of Information and Communication Engineering,
Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon, Gyeonggi-do, 440-746, Korea,
http://www.security.re.kr

Summary
Recently, Sui *et al.* proposed two separable and anonymous ID-based key issuing protocols without secure channel and claimed that their second protocol avoids the key escrow problem. However, in this paper, impersonation attack is proposed to show that Sui *et al.*'s second protocol is not free from the key escrow problem. We also show that their protocol cannot detect the able to access of illegitimate users instead it suffers from the stolen-verifier attack.

Key words:
ID-based cryptography, Key escrow problem, Secure key issuing.

Introduction

An inherent drawback of ID-based cryptosystems is the key escrow problem, *i.e.*, the trusted authority can impersonate a user. To tackle such problem, ID-based key issuing protocols were proposed [1-4]. Recently, in 2005, Sui *et al*. [4] proposed two anonymous ID-based key issuing protocols without secure channel. Their protocols separate two different entities: the authentication and the private key generation can be computed by two different entities Local Registration Authority (LRA) and KGC, respectively. Particularly, they claimed that their second protocol removes the key escrow: in order to avoid the key escrow problem, they support multiple KGCs.

However, in this paper, we point out that Sui *et al.*'s second protocol does not really resolve the key escrow problem by presenting the impersonation attack. We also show that their protocol cannot detect the accesses of illegitimate users. Moreover, we show that their protocol suffers from the stolen-verifier attack.

★ This work was supported by the University IT Research Center Project funded by the Korean Ministry of Information and Communication.
Manuscript received January, 2006.
Manuscript reviced January, 2006.

2. Review of Sui *et al.*'s Second Protocol

In this section, we first recall some definitions and notations and then describe the Sui *et al.*'s second ID-based key issuing protocol.

2.1 Preliminaries

Computational problems. The computational assumptions applied to this paper are based on the Diffie-Hellman problem (DHP). DHP can be classified into as follows:

(1) Discrete Logarithm Problem (DLP): Given a duple of G elements (P, aP), find the integer a.
(2) Computational Diffie-Hellman Problem (CDHP): Given a triple of G elements (P, aP, bP), compute the element $abP \in G$.
(3) Decisional Diffie-Hellman Problem (DDHP): Given a quadruple of G elements (P, aP, bP, cP), decide whether $c = ab$ or not.

Whether G is cyclic group generated by P, whose order is a prime p and a, b and c be elements of Z_p^*. The relationship between above (2) and (3) shows DDHP is solved by CDHP solution, while its reverse is not accomplished despite various efforts up to now. In 2001, Okamoto *et al*. [5] proposed possible existence of signature scheme based on the gap of difficulty between CDHP and DDHP. They defined the Gap Diffie-Hellman (GDH) group to be groups where DDHP is easy but CDHP is hard, and named Gap Diffie- Hellman Problem.

(4) Gap Diffie-Hellman Problem(GDHP): Given a triple of G elements (P, aP, bP), find the element abP with the help of a DDH oracle.

ID-based cryptosystems using a bilinear-pairing are constructed in a GDH group.

Bilinear-pairing. Let G_1 be a cyclic additive group generated by P, whose order is a prime p, and G_2 be a cyclic multiplicative group with the same order p. The bilinear-pairing $e: G_1 \times G_1 \to G_2$ has the following properties:

- Bilinearity: For all $P, Q, R \in G_1$ and $a, b \in Z_p^*$, $e(aP, bQ) = e(P, Q)^{ab}$ or $e(P+Q, R) = e(P, R)e(Q, R)$ and $e(P, Q+R) = e(P, Q)e(P, R)$
- Non-degeneracy: There exists $P, Q \in G_1$, such that $e(P, Q) \neq 1$
- Efficiency: For all $P, Q \in G_1$, there is an efficient algorithm to compute $e(P, Q)$.

2.2 Protocol Description

A user chooses a one-time password after offline authentication by LRA. Then this password together with the identity of user is stored in KGC's databases of "pending private key". With the help of this information, KGC can know the identity associated with the private key which is requested when the user present this one-time password to the KGC. This information also helps the KGC to check the correctness of the "blinded" identity.

The Sui *et al.*'s protocol consists of setup and key generation. The setup procedure is a probabilistic polynomial algorithm, run by KGC that takes a security parameter k and returns system parameters *params* and the master-key. The key generation procedure is a probabilistic polynomial algorithm that takes as input *params*, the KGC's private key and a user's identity $ID \in \{0,1\}^*$; and returns a user private key S_{ID}. The *password* is the user's chosen one-time password during offline authentication and the tuple $(ID, password)$ is stored in KGC1 and KGC2's databases.

(1) Setup. Let $H: \{0,1\}^* \to G_1$ is a one-way hash function. Public information is $I_{SAK1} = \{G_1, G_2, p, e, H, P_{KGC1} = s_1 P, P_{KGC2} = s_2 P\}$, where (s_1, P_{KGC1}) is the private-public key of the first KGC (KGC1) and (s_2, P_{KGC2}) is the private-public key of the second KGC (KGC2). $P_{KGC} = s_1 s_2 P$ is the system public key, and $H(ID)$ is a public key of user A.

(2) Key generation.

1) A: selects a random number r_1.
$A \to KGC1: \ Q_1 = r_1 H(ID), \ T_1 = r_1^{-1} H(password)$

2) KGC1: checks the validity of the request by checking whether $e(Q_1, T_1) = e(H(ID), H(password))$ holds for a certain tuple in KGC1's database.

3) KGC1: computes $s_1 Q_1$ and $s_1 T_1$.
$KGC1 \to A: \ S_1 = s_1 Q_1, \ \sigma_1' = s_1 T_1$.

4) A: verifies the blinded partial private key by checking $e(S_1, P) = e(Q_1, P_{KGC1})$. And verifies the KGC1's signature on the password by $e(\sigma_1', P) = e(T_1, P_{KGC1})$. If both of them hold, A unblinds the encrypted partial private key and the KGC1's blinded signature on the password to obtain the partial private key $K_1 = r_1^{-1} S_1 = s_1 H(ID)$ and KGC1's signature on the password $\sigma_1 = s_1 H(password)$.

5) A: selects a random number r_2.
$A \to KGC2: \ \sigma_1, \ Q_2 = r_2 K_1, \ T_2 = r_2^{-1} H(password)$.

6) KGC2: checks the validity of the request by checking whether $e(Q_2, T_2) = e(H(ID), \sigma_1)$ holds and checks the validity of KGC1's signature by verifying $e(\sigma_1, P) = e(H(password), P_{KGC1})$ where password is obtained from KGC2's database.

7) KGC2: computes $s_2 Q_2$.
$KGC2 \to A: \ S_2 = s_2 Q_2$.

8) A: verifies the blinded private key by checking $e(S_2, P) = e(Q_2, P_{KGC2})$. If it holds, user A unblinds the encrypted partial private key and obtains the final private key $S_{ID} = r_2^{-1} S_2 = s_1 s_2 H(ID)$.

3. Attacks of Sui *et al.*'s Second Protocol

In this section, we point out the weakness of Sui *et al.*'s second protocol by presenting some attacks.

3.1 Impersonation Attack

Key issuing protocols have tackled the key escrow problem in ID-based cryptosystems. In general, a key

issuing protocol reduces the level of trust that needs to be placed in a trusted third party by spreading the multiple trust third parties. In addition, if all of the third parties cooperate they can recover the private key of users, so the key escrow problem arises only if all of the third parties are untrustworthy. However, in this section, we show that the key escrow problem arises if only one of the third parties KGCs is untrustworthy in Sui *et al.*'s key issuing protocol.

[Case 1.] We assume that malicious KGC1 tries to impersonate as a legitimate user A to obtain the private key of user A.

1. KGC1 selects a random number r^*, and computes $\sigma_1 = s_1 H(password)$, $Q_2^* = r^* s_1 H(ID)$ and $T_2^* = r^{*-1} H(password)$ using the tuple stored in his databases, and then sends σ_1 , Q_2^* and T_2^* to KGC2.

2. KGC2 computes $S_2^* = s_2 Q_2^*$ and sends S_2^* to KGC2 without any doubt, since KGC2 does not check the user's identity.

3. Finally, KGC1 can obtain the private key of user A by computing $S_{ID} = r^{*-1} S_2^* = s_1 s_2 H(ID)$.

[Case 2.] We assume that malicious KGC2 tries to impersonate as a legitimate user A to obtain the private key of user A.

1. KGC2 selects a random number r^*, and computes $Q_1^* = r^* H(ID)$ and $T_1^* = r^{*-1} H(password)$ using the tuple stored in his databases, and then sends Q_1^* and T_1^* to KGC1.

2. KGC1 computes $S_1^* = s_1 Q_1^*$ and $\sigma_1^{'*} = s_1 T_1^*$, and sends S_1^* and $\sigma_1^{'*}$ to KGC2 without any doubt, since KGC1 does not check the user's identity.

3. Finally, KGC2 can obtain the private key of user A by computing $S_{ID} = r^{*-1} s_2 S_1^* = s_1 s_2 H(ID)$.

Consequently, malicious KGC can successfully attack the protocol to illegally obtain users' private keys. It means that the key escrow problem is not really solved in their protocol. Though they use a hash value instead of password itself as mentioned in [4], their protocol can still have the key escrow problem since the validity of the request is checked by a hash value of the password.

3.2 Inability to Detect Accesses of Illegitimate Users

In Sui *et al.*'s protocol, an outsider adversary can interrupt execution of the protocol since he/she can also impersonates a legitimate user A. We show that an adversary can hinder user A and KGC from successfully carrying out the protocol as follows:

1. In the first run of the protocol, the adversary can replace the transmitted messages Q_1 and T_1 with $Q_1^* = r^* Q_1$ and $T_1^* = r^{*-1} T_1$ respectively, where r^* is a random number selected by the adversary. Then KGC1 checks the equation $e(Q_1^*, T_1^*) = e(H(ID), H(password))$ then sends $S_1^* = s_1 Q_1^*$ and $\sigma_1^{'*} = s_1 T_1^*$ without any doubt.

2. Similarly, in the fifth run of the protocol, the adversary can replace the transmitted messages Q_2 and T_2 with $Q_2^{'} = r^{'} Q_2$ and $T_2^{'} = r^{'-1} T_2$ respectively, where $r^{'}$ is a random number selected by the adversary. Then KGC2 checks the equation $e(Q_2^{'}, T_2^{'}) = e(H(ID), \sigma_1)$ then sends $S_2^{'} = s_2 Q_2^{'}$ without any doubt.

Consequently, a legitimate user A cannot obtain his private key through the protocol. As mentioned above, these vulnerabilities arise from the fact that KGC does not correctly check the validity of the authentication request messages of user, contrary to Sui *et al.*'s mention. It means that KGCs cannot detect any access of illegitimate users. What is worse, KGCs make no doubt on the access of the adversary even though the adversary intercepts the transmitted messages Q_1 and T_1 and retransmits it. It means that Sui *et al.*'s protocol suffers from the replay attack as part of an impersonation attack. Although the adversary cannot obtain the private key, these attacks give a bad effect on the protocol.

3.3 Stolen-verifier Attacks

Sui *et al.*'s protocol suffers from the stolen-verifier attack if an adversary has the ability to get the stored verifier someway. Suppose that an adversary E has stolen the verifier, *i.e.* tuple $(ID, password)$. It is clear that E can impersonate as a legitimate user of identity ID and can successfully attack the protocol to illegally obtain user's private key. In addition, it is quite probable that the adversary E steals the tuple since the tuple is stored in all databases of KGCs who wish to issue private keys by participating in protocol. Therefore their protocol is vulnerable to the stolen-verifier attack. Despite a hash value instead of a password itself as mentioned in [4],

their protocol can still be broken by stolen-verifier attacks since the validity of the request is checked by a hash value of the password.

4. Conclusion

We proposed some attacks on Sui *et al.*'s second ID-based key issuing protocol: Their protocol is vulnerable to the impersonation attack and the stolen-verifier attack. Also their protocol cannot detect the accesses of illegitimate users. These attacks have led us to the key escrow problem of the protocol, contrary to their statements.

References

[1] D. Boneh, and M. Franklin, ``Identity-based encryption from the Weil pairing," *Advances in Cryptology, Crypto'01*, LNCS 2139, Springer-Verlag, pp. 213-229, 2001.

[2] L. Chen, K. Harrison, N. P. Smart, and D. Soldera, ``applications of multiple trust authorities in pairing-based cryptosystems," *InfraSec 2002*, LNCS 2437, Springer-Verlag, pp. 260-275, 2002.

[3] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang and S. Yoo, ``Secure Key Issuing in ID-Based Cryptography," *in Proc. of the 2nd Australian Information Security Workshop (AISW 2004)*, vol. 26, pp. 66-74, 2004.

[4] A. Sui, S. S. M. Chow, L. C. K. Hui, S. M. Yiu, K. P. Chow, W. W. Tsang, C. F. Chong, K. H. Pun and H. W. Chan, ``Seperable and Anonymous Identity-Based Key Issuing without Secure Channel," *in Proc. of the 11th Intermational Comference on Parallel and Distributed Systems (ICPADS 2005)*, Vol. 2, pp. 275-279, 2005.

[5] T. Okamoto and D. Pointcheval, ``The Gap-Problems: A new class of problems for the security of cryptographic schemes," *4th International Workshop on Practice and Theory in Public Key Cryptosystems (PKC 2001)*, LNCS 1992, Springer-Verlag, preprint, pp. 104-118. 2001.

Biography

Hyunjue Kim received the B.E. and M.S. degrees in Mathematics from Semyung University, Korea, in 1995 and Sogang University, Korea, in 1997 respectively. She received her Ph.D. degree in Engineering from Sungkyunkwan University in 2005, where she is currently Research Professor of School of Information and Communication Engineering. Her interests are on cryptology and information security. At present her research interest focuses on user authentication.

Seungjoo Kim is a professor of School of Information and Communication Engineering of Sungkyunkwan University in Korea. His main research areas are cryptology and information security. He received his B.E., M.E., and Ph.D. degrees in Information Engineering from Sungkyunkwan University in 1994, 1996, and 1999 respectively. He joined KISA (Korea Information Security Agency) in December 1998 and remained until February 2004. In addition, since 2002, he has worked as an IT standard expert on behalf of Korea.

Dongho Won received his B.E., M.E., and Ph.D. degrees from Sungkyunkwan University in 1976, 1978, and 1988, respectively. After working at ETRI (Electronics & Telecommunications Research Institute) from 1978 to 1980, he joined Sungkyunkwan University in 1982, where he is currently Professor of School of Information and Communication Engineering. His interests are on cryptology and information security. Especially, in the year 2002, he was occupied the president of KIISC (Korea Institute of Information Security & Cryptology).