Improvement on Sui et al.'s Separable and Anonymous Key Issuing Protocol in ID-based Cryptosystem

Changji Wang [1,2] **, *Qin Li*** [1] **, *Xingfeng Yang*** [1]
[1] *Department of Computer Science, Guangdong Province Information Security Key Laboratory, Sun Yat-sen University, Guangzhou 510275, P.R.China*
[2] *The State Key Laboratory of Information Security, Graduate School of Chinese Academy of Sciences, Beijing 100039, P. R.China*

Abstract
To avoid the need of secure channel in ID-based public key systems (ID-PKC), Sui et al. proposed a separable and anonymous key issuing protocol in [1]. Recently, R. Gangishetti et al. showed that Sui et al.'s key issuing protocol suffers from the stolen verifier attack and incompetency of KGCs in [2]. However, R.Gangishetti et al. did not give any solutions to resist these attacks. In this paper, we improve Sui et al.'s separable and anonymous key issuing protocol, the improved protocol can resist the stolen verifier attack and remove the incompetency of KGCs.

Key words:
Improve Sui et al.'s separable and anonymous key issuing protocol

1 Introduction

The concept of ID-PKC, proposed by A. Shamir in 1984 [3], allows a user to use any string, such as an email address or IP address that can identify the user, as the public key. Compared with certificate-based public key systems (CA-PKC), ID-PKC is advantageous in key management, since key distribution and key revocation are not required. A sender can send a secure message to a receiver just using the receiver's identity information, even before the receiver obtains his private key from the KGC. But there is an inherent key escrow problem in ID-PKC, i.e., user's private key is known to the KGC. Therefore, the KGC can decrypt any ciphertext and forge signature for any message, so there is no user privacy and authenticity in the system. ID-PKC also requires a secure channel between users and the KGC to deliver private keys. Because of these inherent problems, ID-PKC is considered to be suitable only for small private network with lower security requirements. Therefore providing a secure key issuing mechanism in ID-PKC is an important issue to make the ID-PKC more applicable to the real world.

To tackle the key escrow problem, several proposals have been made using multiple authority approach [6] or using some user-chosen secret information [4], [5]. If the master key of a KGC is distributed to multiple authorities and a private key is computed in a threshold manner [7], key escrow problem of a single KGC can be prevented. However, in many applications multiple identifications of user by multiple authorities are quite a burden. Generating a new private key by adding multiple private keys [6] is another approach, but in this scheme, KGCs have no countermeasure against user's illegal usage. C. Gentry proposed a certificate-based encryption where secure key issuing was provided using some user-chosen secret information [5], but it became a CA-PKC scheme losing the advantage of ID-PKC. S. Sattam et al. successfully removed the necessity of certificate (they named it certificateless public key cryptography) in similar design using user-chosen secret information [4], but their scheme provides only implicit authentication of the public key. The public key securely generated by the user is not certified in any way. Thus any participant using the public key cannot be convinced of whether the public key indeed belongs to the user. Most recently, B. Lee et al. presented a secure key issuing protocol for ID-PKC [8], which sets multiple key privacy authorities (KPAs) in addition to the KGC to protect the privacy of users' private key. The KGC and the KPAs share the original role of the KGC, and they cooperatively compute user's private key. B. Lee et al. claimed that the key escrow problem in ID-PKC had been solved in [8]. However, R. Gangishetti et al. showed that [8] suffered from impersonation, insider attacks and incompetency of KPAs and the key escrow problem remains unsolved [2].

To avoid the need of secure channel in ID-PKC, Sui et al. proposed a separable and anonymous key issuing protocol [1]. Lately, R. Gangishetti et al. showed that Sui et al.'s key issuing protocol suffers from the stolen verifier attack and incompetency of KGCs [2]. However, R. Gangishetti et al. did not give any solutions to resist these attacks [2]. In this paper, we improve Sui et al.'s separable and anonymous key issuing protocol, such that the improved protocol can resist the stolen verifier attack and remove the incompetency of KGCs.

Manuscript received May 5, 2006.
Manuscript revised May 25 , 2006.

2 Background Concepts and Notations

In this Section we briefly review the basic concepts on bilinear pairing and ID-PKC, while introducing notations used in this paper.

2.1 Bilinear pairings and Gap Diffie-Hellman Group

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order q. Let a, b be elements of Z_q^*. We assume that the discrete logarithm problem (DLP) in both G_1 and G_2 are hard. A bilinear pairings is given as $e: G_1 \times G_1 \to G_2$, which satisfies the following properties:

Bilinear: $e(aP, bQ) = e(P, Q)^{ab}$;

Non-degenerate: There exists $P, Q \in G_1$ such that $e(P, Q) \neq 1$;

Computable: There is an efficient algorithm to compute $e(P, Q)$ for all P and $Q \in G_1$

Now we describe some mathematical problems.

Let G be a cyclic multiplicative group generated by g, whose order is a prime q, assume that the inversion and multiplication in G can be computed efficiently. We first introduce the following problems in G.

1. Discrete Logarithm Problem (DLP): Given two elements g and h, to find an integer $x \in Z_q^*$, such that $h = g^x$ whenever such an integer exists.

2. Computation Diffie-Hellman Problem (CDHP): Given $\langle g, g^a, g^b \rangle$ for $a, b \in Z_q^*$ to compute g^{ab}.

3. Decision Diffie-Hellman Problem (DDHP): Given $\langle g, g^a, g^b, g^c \rangle$ for $a, b, c \in Z_q^*$, to decide whether $c \equiv ab \mod q$.

We call G a gap Diffie-Hellman group if DDHP can be solved in polynomial time but there is no polynomial time algorithm to solve DHP with non-negligible probability. Such groups can be found in super-singular elliptic curve or hyper-elliptic curve over finite field, and the bilinear pairings can be derived from the Weil or Tate pairings. For more details, see [4] and [7].

2.2 Notations

Throughout the rest of this paper, we define G_1 be a GDH group and G_2 be a cyclic multiplicative group of the same prime order q, respectively. And define k be a security parameter, P be a generator of G_1 and $e: G_1 \times G_1 \to G_2$ be an admissible bilinear pairing. Besides, we let $H_1(\cdot)$ and $H_2(\cdot)$ be two cryptographic hash functions, where $H_1: \{0,1\}^* \to G_1$ and $H_2: \{0,1\}^* \to Z_q^*$.

3 Review of Sui et al.'s Key Issuing Protocol [1]

A one time password *pwd* can be established between the Local Registration Authority (LRA) and the user after the off-line authentication.

3.1 Setup (run by KGC)

The KGC generates and publishes system parameters $(k, G_1, G_2, P, q, e, H_1, H_2, P_{Pub})$, where $P_{Pub} = sP$ is the system public key and s is the master-key.

3.2 Key Generation

It takes inputs as *params*, master-key s, an arbitrary $ID \in \{0,1\}^*$ and returns a private key S_{ID}. The password *pwd* is user's chosen password during off-line authentication and the tuple $\langle ID, pwd \rangle$ is stored in KGC's database of "pending private key".

1. User $\to$ KGC: User selects a random number r, computes $Q = rH_1(ID)$, $T = r^{-1}H_1(pwd)$, then User sends Q and T to KGC.

2. KGC $\to$ User: KGC checks the validity of the request by checking if $e(Q, T) = e(H_1(ID), H_1(pwd))$ holds for a certain tuple in KGC's database. If it holds, then KGC computes $S = sQ$ and sends S to User.

3. User: User verifies the blinded private key by checking $e(S, P) = e(Q, P_{Pub})$. If it holds, A unblinds the encrypted private key and obtains $sH_1(ID)$.

The user can delete *pwd* after obtaining the private key. The KGC can also remove the tuple $\langle ID, pwd \rangle$ from the database after the protocol.

3.3 Cryptanalysis of Sui et al.'s Key Issuing Protocol

Stolen Verifier Attack: In Sui et al.'s key issuing protocol, $\langle ID, pwd \rangle$ is stored in KGC's database in plaintext. If an adversary steals the database he can have genuine users' secrets on requesting the KGC on behalf of any registered user available in database. Though the KGC stores $\langle ID, pwd \rangle$ for a short-time till the corresponding secret key is issued, it affects the protocol entirely.

Incompetency of KGCs: An adversary can alter the user's requests for private key as follows. The adversary replaces the parameters Q and T with $Q^* = r^* Q$ and $T^* = r^{*-1} T$, respectively. KGC verifies the equality $e(Q^*, T^*) = e(H_1(ID), H_1(pwd))$, then the KGC computes $S^* = sQ^*$ and sends to the user. In this protocol, the KGC cannot check the validity of the parameters received and thus blindly signs on it.

4 Improvements on Sui et al.'s Key Issuing Protocol

A one time password *pwd* can be established between the Local Registration Authority (LRA) and the user after the off-line authentication.

4.1 Setup (run by KGC)

The KGC generates and publishes system parameters $(k, G_1, G_2, P, q, e, H_1, H_2, P_{Pub})$, where $P_{Pub} = sP$ is the system public key and s is the master-key.

4.2 Key Generation

It takes inputs as *params*, master-key s, an arbitrary $ID \in \{0,1\}^*$ and returns a private key S_{ID}. The password *pwd* is user's chosen password during off-line authentication and the tuple $\langle H_1(ID), H_1(pwd), H_2(pwd) \rangle$ is stored in KGC's database of "pending private key".

1. User $\rightarrow$ KGC: User selects a random number r, compute $Q = r^{-1} H_1(ID)$, $T = (r + H_2(pwd)) H_1(pwd)$, then User sends Q and T to KGC.

2. KGC $\rightarrow$ User: KGC checks the validity of the request by checking whether $e(Q,T) = e(H_1(ID), H_1(pwd)) \cdot e(Q, H_1(pwd))^{H_2(pwd)}$ holds for a certain tuple in KGC's database. If it holds, then KGC computes and sends $S = sQ$ to User.

3. User: User verifies the blinded private key by checking $e(S,P) = e(Q, P_{Pub})$. If it holds, User unblinds the encrypted private key and obtains $sH_1(ID)$.

5 Analysis of the improved protocol

5.1 Correctness Analysis

Before KGC make a blind signature on ID for the user, KGC can verify the user by checking the equality

$$\begin{aligned} e(Q,T) &= e(r^{-1}H_1(ID), (r+H_2(pwd))H_1(pwd)) \\ &= e(H_1(ID), H_1(pwd))^{r^{-1}(r+H_2(pwd))} \\ &= e(H_1(ID), H_1(pwd)) \cdot e(H_1(ID), H_1(pwd))^{r^{-1}H_2(pwd)} \\ &= e(H_1(ID), H_1(pwd)) \cdot e(r^{-1}H_1(ID), H_1(pwd))^{H_2(pwd)} \\ &= e(H_1(ID), H_1(pwd)) \cdot e(Q, H_1(pwd))^{H_2(pwd)}. \end{aligned}$$

Also the user is assured of the correctness of his private key using the public keys of KGC and validates the equation $e(S,P) = e(sQ,P) = e(Q,sP) = e(Q, P_{Pub})$.

5.2 Resist the Stolen Verifier Attack

In the improved key issuing protocol, the tuple $\langle H_1(ID), H_1(pwd), H_2(pwd) \rangle$ instead of $\langle ID, pwd \rangle$ in [1] are stored in the KGC's database of "pending private key". According to the one-way property of Hash functions, even if an adversary steals the database he can not have genuine users' secrets on requesting the KGC on behalf of any registered user available in database.

5.3 Remove the Incompetency of KGCs

In the improved key issuing protocol (see Section 4), If an adversary replaces the parameters Q and T with $Q^* = (r^*)^{-1} Q$ and $T^* = r^* T$, respectively. KGC verifies whether the equality

$$e(Q^*, T^*)$$
$$= e(H_1(ID), H_1(pwd)) \cdot e(Q^*, H_1(pwd))^{H_2(pwd)}$$

holds or not. However, the above equation will not be satisfied because

$$e(Q^*, T^*) = e(r^{*-1}Q, r^*T)$$
$$= e(Q, T)$$
$$= e(H_1(ID), H_1(pwd)) \cdot e(Q, H_1(pwd)^{H_2(pwd)})$$
$$\neq e(H_1(ID), H_1(pwd)) \cdot e(Q^*, H_1(pwd)^{H_2(pwd)})$$

So if the adversary makes any changes for Q and T, the KGC can discover it and reject to sign on it. In the improved protocol, the KGC can check the validity of the messages received before he/she blindly signs on them.

5.4 Security Analysis

The security of the improved key issuing protocol is based on the chosen-target CDH assumption and the random oracle model. In fact, the improved key issuing protocol can be regard as the blind signature protocol that the KGC acts the signer. So we can use the similar techniques in [9] to prove the security of the improved key issuing protocol.

Blindness: From the improved protocol (see Section 4), since the blind factor r is chosen randomly from Z_q^*, thus $Q = r^{-1}H_1(ID)$ and $T = (r + H_1(pwd))H_2(pwd)$ are also random elements in the group G_1. An attacker cannot derive r or r^{-1} from Q and T under CDH assumption, thus the attacker can not know who sent the message. Similarly, the attacker cannot obtain $sH_1(ID)$ from sQ, only the legitimate user who knows the blinding parameter can unblind the messages and retrieve the private key.

Unforgeability: This property means that there exists no polynomial-time adversary A with non-negligible advantage Adv(A), where Adv(A) is the probability of A to output l valid message-signature message pairs while the number of invoked blind signing protocols is strictly less than l.

To prove the unforgeability of the blind signature, A.Boldyreva [9] defined the chosen-target CDH assumption for the blind signature and proved an equivalence relation between the unforgeability and chosen-target CDH assumption. The security of the improved key issuing protocol can be proven in a similar way.

Theorem 1. If the chosen-target CDH assumption is valid in G_1, then the improved key issuing protocol is secure against one-more forgery chosen message attack.

5.5 Efficiency Analysis

In the point of efficiency, this improved issuing protocol is less efficient than the original key issuing protocol [1]. KGC can precompute $e(H_1(ID), H_1(pwd))$ in [1], thus when KGC receives the message, KGC just needs to compute one pairing and compares the result with the pre-computed values in the database.

Intuitively, KGC need to compute two pairings and pre-compute one pairing. In fact, we can improve the efficiency as follows. In the step 1 of the key generation stage (see 4.2), the user sends (Q, T), together with $H_2(pwd)$ to KGC. This $H_2(pwd)$ acts as an index such that KGC can choose the right $\langle H_1(ID), H_1(pwd), H_2(pwd) \rangle$ in the database to verify the user.

In addition, KGC stores $\langle H_1(ID), H_1(pwd), H_2(pwd) \rangle$ instead of storing $\langle ID, pwd \rangle$ in [1] to avoid users' $\langle ID, pwd \rangle$ appearing in clear text in the KGC's database.

6 Conclusion

In this paper, we improved the Sui et al.'s separable and anonymous key issuing protocol [1], the proposed protocol can resist stolen verifier and remove the incompetency of key generation centers.

Acknowledgments

We acknowledge the support of the National Natural Science Foundation of China (Grant No. 60503005) and the Natural Science Foundation of Guangdong Province (Grant No.05200302).

References

[1] 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, H. W. Chan, "Seperable and Anonymous Identity-Based Key Issuing without Secure Channel", IACR eprint Archive, Available from http://eprint.iacr.org/2004/322.

[2] R. Gangishetti, M. C. Gorantla, M. L. Das,A. Saxena, "Cryptanalysis of Key Issuing Protocol in ID-Based Cryptosystems", Available from http://arxiv.org/abs/cs.CR/0506015

[3] A. Shamir, “Identity-based cryptosystems and signature scheme”, Advances in Cryptology-Crypto 84, LNCS 196, Springer-Verlag, pp. 47-53, 1984.
[4] S. Sattam, S. Al-Riyami and K. Paterson, “Certificateless public key cryptography”, Advances in Cryptology-Asiacrypt’2003, Springer-Verlag, pp.452-472, 2003.
[5] C. Gentry, “Certificate-based encryption and the certificate revocation problem”, Advances in Cryptology-EUROCRYPT 2003, Springer-Verlag, pp.272-293, 2003.
[6] L. Chen, K. Harrison, N. P. Smart and D. Soldera, “Application of multiple trust authorities in pairing based cryptosystems”, InfraSec 2002, Springer-Verlag, pp.260-275, 2002.
[7] D. Boneh and F. Franklin, “Identity-based encryption from the Wail pairing”, Advances in Cryptology-Crypt’2001, Springer-Verlag, pp.213-229, 2001.
[8] B. Lee, E. Boyd, E. Daeson, K. Kim, J. Yang and S. YooS, “Secure key issuing in ID-based cryptography”, In proceedings of the Second Australian Information Security Workshop-AISW 2004, pp.69-74, 2004.
[9] A.Boldyreva, “Efficient Threshold Signature, Multisignature, and Blind Signature Schemes based on the Gap Diffie-Hellman Group Signature Scheme”, Prodeedings of Public Key cryptography-PKC2003, LNCS2567, Springer-Verlag, pp.31-46, 2003.

Changji Wang received the M.S. degree in Applied Mathematics from Sun Yat-sen University in 1997, and received the Phd. degree in Applied Mathematics from the Graduate School of Chinese Academy of Sciences. During 2002-2004, he stayed in network research center of Tsinghua University for postdoctoral research. He now is a teacher at the department of computer science in Sun Yat-sen Unversity.

Qin Li received the B.S degree in Hunan Normal Univercity in 2005. She now is studying in the Department of Computer Science, Guangdong Province Information Security Key Laboratory, Sun Yat-sen University.

Xingfeng Yang received the B.S degree in Shandong Normal Univercity in 2005. She now is studying in the Department of Computer Science, Guangdong Province Information Security Key Laboratory, Sun Yat-sen University.