2009

Security vulnerability of ID-based key sharing schemes

Jung Yeon Hwang
Electronics and Telecommunications Research Institute, Korea

Taek-Young Youn
Korea University - Korea

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

Security vulnerability of ID-based key sharing schemes

Abstract

Recently, several ID-based key sharing schemes have been proposed, where an initiation phase generates users' secret key associated with identities under the hardness of integer factorization. In this letter, we show that, unfortunately any key sharing scheme with this initiation phase is intrinsically insecure in the sense that the collusion of some users enables them to derive master private keys and hence, generating any user's secret key.

Keywords

sharing, schemes, id, vulnerability, key, security

Publication Details

Hwang, J., Youn, T. & Susilo, W. (2009). Security vulnerability of ID-based key sharing schemes. IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences, E92-A (10), 2641-2643.

LETTER

Security Vulnerability of ID-Based Key Sharing Schemes

JungYeon HWANG[†a)], Taek-Young YOUN[††b)], *Nonmembers, and* Willy SUSILO[†††c)], *Member*

SUMMARY Recently, several ID-based key sharing schemes have been proposed, where an initiation phase generates users' secret key associated with identities under the hardness of integer factorization. In this letter, we show that, unfortunately any key sharing scheme with this initiation phase is intrinsically insecure in the sense that the collusion of some users enables them to derive master private keys and hence, generating any user's secret key.

key words: *ID-based key exchange, collusion attack*

1. Introduction

Sharing a common key among users and obtaining confidence of key-sharing only between intended participants are essential for secure network communication. Since the seminal concept of conference key distribution in [3], much research has been conducted to improve the efficiency while achieving stronger security properties. Combining identity(ID)-based cryptography [6] with a key sharing scheme provides a benefit since (certificate-based) public key management procedures are simplified in the sense that an arbitrary string such as an IP or e-mail address may be used as a public key.

Among various approaches for ID-based key sharing (IBKS), several schemes [4], [5], [7] have incorporated an initiation phase that generates a secret key for an identity under the hardness of integer factorization of a RSA-modulus. In order to treat an n-bit identity, i.e., 2^n identities, the initiation phase makes use of n master secret values and their corresponding (global) public commitments. A secret key for an n-bit identity is generated as a linear combination of the n master secret values and the n bits of the identity under a secret modulus. It has been believed that the initiation phase securely works under the hardness of integer factorization of the underlying RSA-modulus. (Note that a similar discrete logarithm based initiation phase in a group of prime order is obviously insecure because the inverse of an exponent can be easily computed under the known order.)

In this letter, unfortunately, we show that any IBKS scheme that uses this specific initiation phase is totally insecure against collusion attacks. That is, a reasonable number of colluding users can efficiently compute master private keys and so any user's secret key. To show this, we first review the recent IBKS scheme in [4] that uses this initiation phase. Then, we concretely present an efficient method to compute partial information on two prime factors of the modulus. Extending this method, we present a method to compute all master secret keys. Additionally we show that the partial information can be used for a chairperson impersonation attack on the IBKS scheme. This impersonation attack is also applicable to the schemes proposed in [5], [7] that incorporate a similar key distribution phase.

2. Jeong et al.'s IBKS Scheme

We briefly recall the IBKS scheme in [4].

Initiation Phase. The trusted center (TC) chooses two large prime numbers, p and q, and generates an arbitrary prime e and an integer d such that $N = pq$, $\lambda = lcm(p-1, q-1)$, and $ed = 1 \pmod{\lambda}$. TC picks g which is a primitive element of $GF(p)$ and $GF(q)$, and a random vector $X = (x_1, \ldots, x_n)$ where $2 \le x_i \le \lambda - 1$ for $1 \le i \le n$. TC constructs $G = (g^{x_1}, \ldots, g^{x_n})$. Let f be a one-to-one and one-way function which maps an identity ID_i of a user U_i to an n-dimensional vector, $f(ID_i) = (\alpha_{i1}, \ldots, \alpha_{in})$, $\alpha_{ij} \in \{0, 1\}$ for $1 \le j \le n$. Let $A \circ B$ denote the inner product of two vectors A and B. For the user U_i, TC computes u_i, v_i, and s_i satisfying (1) $u_i = X \circ f(ID_i) = \alpha_{i1}x_1 + \cdots + \alpha_{in}x_n \pmod{\lambda}$, (2) $v_i = u_i^{-1} \pmod{\lambda}$, (3) $s_i = ID_i^d \pmod{N}$, and sends (s_i, v_i) to each U_i over a secure channel. TC keeps $(p, q, \lambda, X, d, u_i)$ secret and publishes $(N, g, f(\cdot), h(\cdot), G, e)$ to all users.

Key Distribution Phase. Let U_0 be a chairperson and $U_1, \ldots, U_m$ want to participate in a conference. They proceed as follows: (1) U_i generates a random r_i, computes $W_i = g^{r_i} \pmod{N}$, and sends (ID_i, W_i) to U_0. (2) U_0 generates two random r, R, and computes $C_0 = s_0 g^{rh(t)} \bmod N$, $e' = H(C_0, t, W_0)$ and $W_0 = g^R$ where t is a time-sequence and $H(\cdot)$ is a collision resistant hash function. (3) For each U_i, U_0 computes $f(ID_i) = (\alpha_{i1}, \ldots, \alpha_{in})$, $G_i = \prod_{j=1}^{n} (g^{x_j})^{\alpha_{ij}} = g^{\mu_i} \pmod{N}$ and $C_i = G_i^{ree'} W_i^R \pmod{N}$. (4) U_0 sends $\{C_0, C_i, t, W_0\}$ to each U_i $(i = 1, \ldots, m)$ and computes $CK = kdf(g^{ree'})$ with a key derivation function

Manuscript received February 12, 2009.
Manuscript revised April 28, 2009.

[†]The author is with Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea.
[††]The author is with the Graduate School of Information Management and Security, Korea University, Seoul, Korea.
[†††]The author is with the Centre for Computer and Information Security Research, School of Computer Science and Software Engineering, University of Wollongong, Australia.
a) E-mail: videmot@etri.re.kr
b) E-mail: taekyoung@korea.ac.kr
c) E-mail: wsusilo@uow.edu.au

DOI: 10.1587/transfun.E92.A.2641

$kdf(\cdot)$.

Key Recovery Phase. On receiving $\{C_0, C_i, t, W_0\}$ from U_0, each U_i computes $C_{0i} = C_i[(W_0)^{r_i}]^{-1} \pmod N$, checks a given time sequence t and $ID_0^{e'} \stackrel{?}{=} C_0^{ee'} \cdot [(C_{0i})^{h(t)v_i}]^{-1} \pmod N$. U_i computes the conference key $CK = kdf(C_{0i}^{r_i}) = kdf(g^{ree'})$.

3. Security Vulnerability

In this section we show that the initiation phase is vulnerable to collusion attacks. The weakness is mainly caused by the structure of a user secret key computed by linearly combining n *static* master secret keys and n bits of an identity under the secret modulus λ. In particular we show that a multiple of λ, λ, and other master private keys can be revealed with an overwhelming probability.

Computation of a Multiple of λ. Assume that $\{(v_1, s_1), \ldots, (v_{n+1}, s_{n+1})\}$ is a set of $n+1$ user secret keys. Since $v_i = u_i^{-1} \pmod \lambda$, $v_i u_i = t_i\lambda + 1$ holds for non-zero $t_i \in \mathbb{Z}$. Let $f(ID_i) = (\alpha_{i,1}, \ldots, \alpha_{i,n})$, $u_i = \alpha_{i,1}x_1 + \cdots + \alpha_{i,n}x_n$, $A_i = (v_i\alpha_{i,1}, \ldots, v_i\alpha_{i,n})$, and $X = (x_1, \ldots, x_n)$. We have $v_i(\alpha_{i,1}x_1 + \cdots + \alpha_{i,n}x_n) = A_i \circ X = t_i\lambda + 1$. We compute $n+1$ integers $c_1, \ldots, c_{n+1}$ such that $\sum_{i=1}^{n+1} c_i A_i = 0_n (= (0, \ldots, 0))$ and some c_i is non-zero. As shown in the analysis of our attack, this computation can be efficiently done. Note that

$$\begin{aligned}
&\sum_{i=1}^{n+1} c_i(A_i \circ X) = \sum_{i=1}^{n+1} c_i(t_i\lambda + 1) \\
&\Leftrightarrow \sum_{i=1}^{n+1}(c_iA_i) \circ X = \sum_{i=1}^{n+1} c_it_i\lambda + \sum_{i=1}^{n+1} c_i \\
&\Leftrightarrow 0_n \circ X = \sum_{i=1}^{n+1}(c_it_i)\lambda + \sum_{i=1}^{n+1} c_i \\
&\Leftrightarrow \sum_{i=1}^{n+1} c_i = \left(\sum_{i=1}^{n+1}(-c_it_i)\right)\lambda.
\end{aligned}$$

Hence, we can compute $\sum_{i=1}^{n+1} c_i$, which is a multiple of λ.

Using a multiple of λ, we particularly show that an chairperson-impersonation attack can be mounted on the scheme. Assume that an adversary $\mathcal{F}$ obtains $t\lambda$ for $t \in \mathbb{Z}$. Let $w = t\lambda/gcd(e, t\lambda)$. Since $gcd(e, \lambda) = 1$, $w = t'\lambda$ where $t' = t/gcd(e, t)$. The adversary computes d' such that $d'e = 1 \pmod w$. Note that $d'e = 1 \pmod \lambda$ and $d = d' \pmod \lambda$ because $d'e = t''t'\lambda + 1$ for some $t'' \in \mathbb{Z}$ and $gcd(e, \lambda) = 1$. Therefore, $\mathcal{F}$ can compute $s_j = ID_j^d = ID_j^{d'} \pmod N$ for U_j. In the key distribution phase, $\mathcal{F}$ with the key $s_0 = ID_0^d$ can easily impersonate the chairperson U_0.

Computation of Master Secret Keys. We show that one can highly compute more important master secret keys using several multiples of λ.

First, we present a simple algorithm to compute λ using multiples of λ which are computed in the above method. λ cannot be directly computed from $(\sum_{i=1}^{n+1}(-c_it_i)) \cdot \lambda$ because $t_1, \ldots, t_n$ are unknown values in $(\sum_{i=1}^{n+1}(-c_it_i)) \cdot \lambda$. However, with two independent and random multiples of λ obtained in Sect. 3, we can determine λ with an overwhelming probability. Suppose that $\hat{t}_1\lambda$ and $\hat{t}_2\lambda$ are given. Then, $gcd(\hat{t}_1\lambda, \hat{t}_2\lambda) = \hat{t}\lambda$ where $\hat{t} = gcd(\hat{t}_1, \hat{t}_2)$. It is well-known that the probability that two random numbers are relatively prime is $6/\pi^2 \approx 0.6$ [8]. Applying this fact to our case we obtain $\Pr[\hat{t} = 1] \approx 0.6$ assuming $\hat{t}_1$ and $\hat{t}_2$ are random. If we iterate the above process with more multiples of λ we can obtain λ more correctly. Therefore we expect to extract λ within a few iterations.

Next we show that other master secret keys $x_1, \ldots, x_n$ can be derived. First we compute $u_i = v_i^{-1} \pmod \lambda$ for all $i = 1, \ldots, n$. By construction, $u_i = \alpha_{i,1}x_1 + \cdots + \alpha_{i,n}x_n \pmod \lambda$ where $f(ID_i) = (\alpha_{i,1}, \ldots, \alpha_{i,n})$ and $\alpha_{ij} \in \{0, 1\}$. Then, $x_1, \ldots, x_n$ are computed by

$$\begin{pmatrix} \alpha_{1,1} & \cdots & \alpha_{1,n} \\ \vdots & & \vdots \\ \alpha_{n,1} & \cdots & \alpha_{n,n} \end{pmatrix}\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}. \quad (1)$$

Assume that the inverse of the above coefficient matrix exists. If it does not exist, we can collect other user secret keys until the inverse exists. Since the entries α_{ij} of the matrix are all publicly known, such a collection is efficiently constructed. Moreover, for security, the scheme should use strong primes $p = 2p' + 1$ and $q = 2q' + 1$ where p' and q' are primes. In this case, $\lambda = lcm(p-1, q-1) = 2p'q'$ and so the inverse highly exists. Obviously, using λ and $(x_1, \ldots, x_n)$, anyone can generate a secret key for an identity.

Analysis of Our Attack. We show that our attack is efficient. In order to obtain a multiple of λ, our first method should compute $c_1, \ldots, c_{n+1}$ which satisfy

$$\sum_{i=1}^{n+1} c_iv_i(\alpha_{i,1}, \ldots, \alpha_{i,n}) = 0_n. \quad (2)$$

This can be efficiently done as follows. Let $(z_1, \ldots, z_{n+1})$ be a solution for $\sum_{i=1}^{n+1} z_i(\alpha_{i,1}, \ldots, \alpha_{i,n}) = 0_n$. Then $c_i = z_i \prod_{j=1, j\neq i}^{n+1} v_j$ satisfies Eq. (2), because $\sum_{i=1}^{n+1} c_iv_i\vec{\alpha}_i = \sum_{i=1}^{n+1}(z_i \prod_{j=1, j\neq i}^{n+1} v_j)v_i\vec{\alpha}_i = (\prod_{j=1}^{n+1} v_j)(\sum_{i=1}^{n+1} z_i\vec{\alpha}_i) = 0_n$ where $\vec{\alpha}_i = (\alpha_{i,1}, \ldots, \alpha_{i,n})$. Now we present an efficient method to find $(z_1, \ldots, z_{n+1})$. Assume that without loss of generality, z_{n+1} is fixed. Let's consider the $n \times n$ binary coefficient matrix in the following relation,

$$\begin{pmatrix} \alpha_{1,1} & \cdots & \alpha_{n,1} \\ \vdots & & \vdots \\ \alpha_{1,n} & \cdots & \alpha_{n,n} \end{pmatrix}\begin{pmatrix} z_1 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} -z_{n+1}\alpha_{n+1,1} \\ \vdots \\ -z_{n+1}\alpha_{n+1,n} \end{pmatrix}.$$

Using well-known Gaussian Elimination method [2] on the matrix one can efficiently find $(z'_1, \ldots, z'_n)$ such that $z'_i = \frac{\gamma_i}{\beta_i}z_{n+1}$ for $\gamma_i, \beta_i \in \mathbb{Z}$. Let $\beta = lcm(\beta_1, \ldots, \beta_n)$. If $z_{n+1} = \beta m$ for some $m \in \mathbb{Z}$ is selected then $z_i = z'_i \cdot z_{n+1} = \frac{\gamma_i}{\beta_i} \cdot \beta m$ is an integer. Therefore $(z_1 = \frac{\gamma_1}{\beta_1} \cdot \beta m, \ldots, z_n = \frac{\gamma_n}{\beta_n} \cdot \beta m, z_{n+1} = \beta \cdot m)$ is

a solution for $\sum_{i=1}^{n+1} z_i(\alpha_{i,1}, ..., \alpha_{i,n}) = 0_n$ as desired. For clarification, let's consider a simple example with $A_1 = (0,1,1)$, $A_2 = (1,0,1)$, $A_3 = (1,1,0)$, and $A_4 = (1,1,1)$. Our goal is to find four integers z_1, z_2, z_3, and z_4 such that $z_1A_1+z_2A_2+z_3A_3+z_4A_4 = (z_2+z_3+z_4, z_1+z_3+z_4, z_1+z_2+z_4) = 0_n$. Let's consider the equivalent matrix representation,

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} -z_4 \\ -z_4 \\ -z_4 \end{pmatrix}.$$

We can obtain $z_1 = z_2 = z_3 = \frac{-z_4}{2}$ using Gaussian Elimination method. For a given $z_4 = 2$ it is obvious that $(z_1 = -1, z_2 = -1, z_3 = -1, z_4 = 2)$ is a solution for the above relation. Selecting multiples of 2 for z_4 many solutions can be obtained.

There may exist lots of solutions for Eq. (2) as shown above. Intuitively, this is because we consider $n + 1$ n-dimensional vectors, which are linearly dependent. These different solutions can be used for computing different multiples of λ. (If necessary, we can select one more user secret key to construct a new collection of $n + 1$ user secret keys and compute multiples of λ.) Our second method may use these multiples to obtain λ by computing the great common divisor(GCD) of the multiples. Note that, as mentioned previously, the computation of multiples is efficiently conducted and also is GCD. Alternatively, one can use the method in [1] with a multiple of λ for computing p, q, and $\lambda = lcm(p-1, q-1)$.

Our attack mainly relies on the parameter n for the size of a (transformed) identity, i.e., $f(ID) = (\alpha_1, \ldots, \alpha_n)$ where $\alpha_j \in \{0,1\}$. One may use an arbitrary identity such as a phone number, an IP address, or an e-mail address[†]. For a promising example, when an IP address is used as identity, $n = 32$ and $n = 128$ are fixed for IPv4 and IPv6, respectively. Since f is one-to-one, $f(ID)$ may be a 32-bit or 128-bit string. In this case our attack just needs collusion of 33 or 129 users and so quite practical. Generally, for total $W = 2^n$ users or identities, our attack only requires that $\log W(= n)$ users collude, i.e., the polynomial number of colluding users with respect to security parameter. This means that our attack is quite reasonable.

The IBKS schemes in [5], [7] uses the initiation phase of [4] and the same system parameters $(N, g, f(\cdot), h(\cdot), G, e)$ (except using an additional hash function[††]). The main ides of our attack is to exploit only weaknesses in the user key generation structure of the initiation phase. Our analysis, which is based on the initiation phase, is also applied for any key sharing scheme with this initiation phase. In addition, the chairperson U_0 in the IBKS schemes of [5], [7] also executes a similar key distribution phase, using only its private key $s_0 = ID_0^d$ as in [4]. Our chairperson impersonation attack is still similarly applied to the schemes in [5], [7].

[†]In [4], [5], [7], there is neither explicit definition of user identity nor clear choice of n.

[††]This additional hash function is independently used regardless of our attack.

4. Conclusion

In this letter, we demonstrated security vulnerability in ID-based key sharing scheme against collusion attacks. In particular, we showed that any ID-based key sharing scheme that incorporates an initiation phase to generate users' secret key associated with their identities is intrinsically insecure against collusion attacks. We demonstrated our argument with an attack against the scheme in [4]. It is not clear if our attack can be applied to other key sharing schemes using a different initiation phase with RSA parameters.

Acknowledgement

This work was supported by the IT R&D program of MKE/KEIT. [2008-F-036-02, Development of Anonymity-based u-knowledge Security Technology]

References

[1] D. Boneh, "Twenty years of attacks on the RSA cryptosystem," in Notices of the AMS, pp.203–213, 1999.
[2] R.W. Farebrother, "Linear least squares computations," in STATISTICS: Textbooks and Monographs, Marcel Dekker Inc., p.12, 1988.
[3] I. Ingemaresson, T.D. Tand, and C.K. Wong, "A conference key distribution system," IEEE Trans. Inf. Theory, vol.28, no.5, pp.714–720, 1982.
[4] I.R. Jeong, J.O. Kown, and D.H. Lee, "On the strong forward secrecy of the improved Chikazawa-Yamagishi ID-based key sharing," IEICE Trans. Fundamentals, vol.E90-A, no.11, pp.2626–2628, Nov. 2007.
[5] I.R. Jeong, J.O. Kwon, and D.H. Lee, "Strong ID-based key distribution," IEICE Trans. Commun., vol.E91-B, no.1, pp.306–308, Jan. 2008.
[6] A. Shmair, "Identity-based cryptosystems and signature schemes," Proc. Crypto 84, pp.47–53, 1984.
[7] Y. Tseng and J. Jan, "Improvement of Chikazawa-Yamagishi ID-based key sharing system," Electron. Lett., vol.34, no.12, pp.1221–1222, 1998.
[8] J.E. Nymann, "On the probability that positive integers are relatively prime," J. Number Th., vol.4, pp.469–473, 1972.