Sensors **2015**, *15*, 25033-25038; doi:10.3390/s151025033

OPEN ACCESS
sensors
ISSN 1424-8220
www.mdpi.com/journal/sensors

Comment

A Note on an Improved Self-Healing Group Key Distribution Scheme

Hua Guo [1,2,*], Yandong Zheng [2], Biao Wang [3] and Zhoujun Li [1,2]

[1] State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China; E-Mail: lizj@buaa.edu.cn

[2] Beijing Key Laboratory of Network Technology, Beihang University, Beijing 100191, China; E-Mail: zy1406423@buaa.edu.cn

[3] School of Information Science and Technology, University of International Relations, Beijing 100091, China; E-Mail: wangbiao@uir.edu.cn

* Author to whom correspondence should be addressed; E-Mail: hguo@buaa.edu.cn; Tel./Fax: +86-108-233-8247.

Academic Editor: Leonhard M. Reindl

Received: 23 June 2015 / Accepted: 16 September 2015 / Published: 29 September 2015

Abstract: In 2014, Chen *et al.* proposed a one-way hash self-healing group key distribution scheme for resource-constrained wireless networks in the journal of Sensors (14(14):24358-24380, doi: 10.3390/ s141224358). They asserted that their Scheme 2 achieves mt-revocation capability, mt-wise forward secrecy, any-wise backward secrecy and has mt-wise collusion attack resistance capability. Unfortunately, this paper pointed out that their scheme does not satisfy the forward security, mt-revocation capability and mt-wise collusion attack resistance capability.

Keywords: self-healing group key distribution; forward security; backward secrecy; collusion attack

---

1. Introduction

Group communication includes a group manager (GM) and some group members, in which all of the group members share a common session key which is distributed by GM. In order to achieve secure group communication in unreliable wireless networks, Staddon *et al.* [1] introduced a group key distribution scheme with self-healing mechanism, which allows a group member to recover session

keys even if he doesn't receive the corresponding broadcast messages because of packet loss, without requesting anything to the group manager. Recently, Chen *et al.* [2] developed two schemes to realize the self-healing group key distribution based on one-way hash chain. The proposed Scheme 2 has the constant storage overhead and low communication overhead, thus is very suitable for the resource-constrained wireless networks. They assert that their scheme is secure, *i.e.*, satisfies mt-revocation capability, mt-wise forward secrecy, any-wise backward secrecy and resistance to mt-wise collusion attack. Unfortunately, we found a revoked user can recover other legitimate users' personal secrets which can be used to recover the current session's session key, this directly breaks the forward security, mt-revocation capability and mt-wise collusion attack resistance capability. Thus, Chen *et al.*'s Scheme 2 is insecure.

2. Overview of Chen *et al.*'s Scheme

Chen *et al.*'s self-healing group key distribution Scheme 2 includes five parts: Set up, Broadcast in session j, Group session key recovery and self-healing, Group member addition and Group member revocation. Here we only describe the first three parts which is helpful to understand the attack.

(1) Set up

The GM selects a random $2t$-degree polynomial $s_1(x) = a_0 + a_1x + \cdots + a_{2t}x^{2t}$ and a random t-degree polynomial $s_2(x) = b_0 + b_1x + \cdots + b_tx^t$ from $F_q[x]$. Then, the GM chooses a random value ε_1 from F_q. The GM sends the user's personal secret $\mathcal{S}_i = \{\varepsilon_1 \cdot s_1(i), \varepsilon_1 \cdot s_2(i)\}$ to a user via a secure channel.

(2) Broadcast in session j (for $1 \leq j \leq m$)

Let $\mathbf{R}_j = \{R_j^1, R_j^2, \cdots, R_j^{j'}, \cdots, R_j^j\}$ be the set of revoked users before and in session j, where $R_j^{j'}$ is the set of users who join the group in session j' and are revoked before and in session j. $R_j^{j'} = \{U_{r_1^{j'}}, U_{r_2^{j'}}, \cdots, U_{r_{w_{j'}}^{j'}}\}$ and $|R_j^{j'}| = w_{j'} \leq t$. $r_1^{j'}, r_2^{j'}, \cdots, r_{w_{j'}}^{j'}$ are the IDs of users in $R_j^{j'}$. $R_j^{j'} = \emptyset$ if no users joined the group in session j'.

- The GM chooses a random value $k_j^0 \in F_q$ and a one-way hash function $h(\cdot)$. Note that $h^i(\cdot)$ denotes applying i times hash operation. Then GM constructs the j-th key chain for session j: $\{k_j^1, k_j^2, \cdots, k_j^j\}$, where

$$\begin{aligned} k_j^1 &= h(k_j^0) \\ k_j^2 &= h(k_j^1) = h(h(k_j^0)) = h^2(k_j^0) \\ &\cdots, \\ k_j^j &= h(k_j^{j-1}) = h(h(k_j^{j-2})) = \cdots = h^j(k_j^0) \end{aligned}$$

For security, $k_j^0 (1 \leq j \leq m)$ is different from each other.

The GM splits the $k_j^{j'}$ into two t-degree polynomials, $U_j^{j'}(x)$ and $V_j^{j'}(x)$, where

$$k_j^{j'} = U_j^{j'}(x) + V_j^{j'}(x), j' = 1, 2, \cdots, j$$

– To construct the revocation polynomials for session j, the GM firstly chooses number sets $\overline{R}_j^{j'}$, where $\overline{R}_j^{j'} = \{\overline{r}_1^{j'}, \overline{r}_2^{j'}, \cdots, \overline{r}_{t-w_{j'}}^{j'}\}$ are random numbers which are not used as a user ID and different from each other. Then, the GM computes

$$A_j^{j'}(x) = \Pi_{z=1}^{|R_j^{j'}|}(x - r_z^{j'})\Pi_{z'=1}^{t-|R_j^{j'}|}(x - \overline{r}_{z'}^{j'}), j' = 1, 2, \cdots, j$$

– The GM chooses a random session key K_j from F_q. Then, the GM computes

$$M_j^{j'}(x) = A_j^{j'}(x) \cdot U_j^{j'}(x) + \varepsilon_{j'} \cdot s_1(x)$$

and

$$N_j^{j'}(x) = V_j^{j'}(x) + \varepsilon_{j'} \cdot s_2(x)$$

After that, the GM broadcasts the message

$$\begin{aligned} B_j \quad = \quad & \mathbf{R}_j \cup \overline{\mathbf{R}}_j \cup \{M_j^{j'}(x) | j' = 1, 2, \cdots, j\} \cup \{N_j^{j'}(x) | j' = 1, 2, \cdots, j\} \\ & \cup \{E_{k_j^{j'}}(K_{j'}) | j' = 1, 2, \cdots, j\} \end{aligned}$$

where $\overline{\mathbf{R}}_j = \{\overline{R}_j^1, \overline{R}_j^2, \cdots, \overline{R}_j^j\}$ and $E_k(\cdot)$ is a symmetric encryption function.

(3) Group session key recovery and self-healing
Any legitimate user $U_i \in G_j^{j'}$ can recover the j-th session key when he receives the broadcast message B_j as follows.

– U_i uses his personal secret $\varepsilon_{j'} \cdot s_1(i)$ and $\varepsilon_{j'} \cdot s_2(i)$ to compute

$$U_j^{j'}(i) = \frac{M_j^{j'}(i) - \varepsilon_{j'} \cdot s_1(i)}{A_j^{j'}(i)}$$

and

$$V_j^{j'}(i) = N_j^{j'}(i) - \varepsilon_{j'} \cdot s_2(i)$$

Then, U_i computes $k_j^{j'} = U_j^{j'}(i) + V_j^{j'}(i)$.

– U_i uses the hash function $h(\cdot)$ to compute all $\{k_j^{j''}\}$ for $j' < j'' \leq j$ in the j-th key chain.

– U_i recovers the session keys $\{K_{j''}\}(j' < j'' \leq j)$ by decrypting $E_{k_j^{j''}}(K_{j''})$ $(j' < j'' \leq j)$ with corresponding keys $\{k_j^{j''}\}(j' < j'' \leq j)$.

3. Cryptanalysis of Chen *et al.*'s Scheme 2

In this section we exhibit the attack on Chen *et al.*'s Scheme 2 step by step, and explain why this attack exists.

3.1. Attack on Chen et al.'s Scheme 2

Let $G_{j_1}^{j'}$ denote the users who join the group in session j' and are still legitimate in session j_1 where $j' < j_1$. Suppose that $U_i \in G_{j_1}^{j'}$ and U_i is revoked in session $j_2 (j' < j_1 < j_2)$. Now we are ready to show how U_i, who is revoked in session j_2, recovers the personal secret of another user who is legitimate in session j_2, furthermore uses this personal secret to compute the session key K_{j_2} which should be kept secret from U_i.

Step 1. U_i computes $k_{j'}^{j'}$ and $k_{j_1}^{j'}$ with his personal key $\mathcal{S}_i$ and the broadcast messages $M_{j'}^{j'}(x)$, $N_{j'}^{j'}(x)$ and $M_{j_1}^{j'}(x)$, $N_{j_1}^{j'}(x)$.

Step 2. In session j', U_i receives the broadcast messages $M_{j'}^{j'}(x), N_{j'}^{j'}(x)$, where

$$M_j^{j'}(x) = A_j^{j'}(x) \cdot U_j^{j'}(x) + \varepsilon_{j'} \cdot s_1(x) \tag{1}$$

and

$$N_j^{j'}(x) = V_j^{j'}(x) + \varepsilon_{j'} \cdot s_2(x) \tag{2}$$

Note that $k_{j'}^{j'} = U_{j'}^{j'}(x) + V_{j'}^{j'}(x)$, Equation (2) can be converted to $N_j^{j'}(x) = k_{j'}^{j'} - U_j^{j'}(x) + \varepsilon_{j'} \cdot s_2(x)$. Let Equation (1) + $A_j^{j'}(x) \cdot Equation(2)$, U_i can obtain

$$M_{j'}^{j'}(x) + A_{j'}^{j'}(x) \cdot N_{j'}^{j'}(x) = k_{j'}^{j'} \cdot A_{j'}^{j'}(x) + \varepsilon_{j'} \cdot s_1(x) + A_{j'}^{j'}(x) \cdot \varepsilon_{j'} \cdot s_2(x) \tag{3}$$

With the values of $k_{j'}^{j'}$ which is computed from step (1), U_i can obtain

$$M_{j'}^{j'}(x) + A_{j'}^{j'}(x) \cdot N_{j'}^{j'}(x) - A_{j'}^{j'}(x) \cdot k_{j'}^{j'} = \varepsilon_{j'} \cdot s_1(x) + A_{j'}^{j'}(x) \cdot \varepsilon_{j'} \cdot s_2(x) \tag{4}$$

Step 3. Since U_i is legitimate in session j_1, U_i can obtain the similar result in the same way:

$$M_{j_1}^{j'}(x) + A_{j_1}^{j'}(x) \cdot N_{j_1}^{j'}(x) - A_{j_1}^{j'}(x) \cdot k_{j_1}^{j'} = \varepsilon_{j'} \cdot s_1(x) + A_{j_1}^{j'}(x) \cdot \varepsilon_{j'} \cdot s_2(x) \tag{5}$$

Let Equation (4) – Equation (5), user U_i can obtain

$$\begin{aligned} &M_{j'}^{j'}(x) + A_{j'}^{j'}(x) \cdot N_{j'}^{j'}(x) - A_{j'}^{j'}(x) \cdot k_{j'}^{j'} - M_{j_1}^{j'}(x) - A_{j_1}^{j'}(x) \cdot N_{j_1}^{j'}(x) + A_{j_1}^{j'}(x) \cdot k_{j_1}^{j'} \\ =&(A_{j'}^{j'}(x) - A_{j_1}^{j'}(x)) \cdot \varepsilon_{j'} \cdot s_2(x) \end{aligned} \tag{6}$$

Step 4. U_i computes $\varepsilon_{j'} \cdot s_2(x)$ as

$$\begin{aligned} &\varepsilon_{j'} \cdot s_2(x) \\ =&\frac{M_{j'}^{j'}(x) + A_{j'}^{j'}(x) \cdot N_{j'}^{j'}(x) - A_{j'}^{j'}(x) \cdot k_{j'}^{j'} - M_{j_1}^{j'}(x) - A_{j_1}^{j'}(x) \cdot N_{j_1}^{j'}(x) + A_{j_1}^{j'}(x) \cdot k_{j_1}^{j'}}{(A_{j'}^{j'}(x) - A_{j_1}^{j'}(x))} \end{aligned} \tag{7}$$

Take $\varepsilon_{j'} \cdot s_2(x)$ to Equation (3), U_i computes $\varepsilon_{j'} \cdot s_1(x)$ as

$$\varepsilon_{j'} \cdot s_1(x) = M_{j'}^{j'}(x) + A_{j'}^{j'}(x) \cdot N_{j'}^{j'}(x) - A_{j'}^{j'}(x) \cdot k_{j'}^{j'} - A_{j'}^{j'}(x) \cdot \varepsilon_{j'} \cdot s_2(x) \tag{8}$$

Step 5. U_i gets a legitimate user's identity, v, in session j_2 by observing $R_j^{j'}$ where $j > j_2$.

Step 6. U_i computes $\varepsilon_{j'} \cdot s_1(v)$ and $\varepsilon_{j'} \cdot s_2(v)$ through $\varepsilon_{j'} \cdot s_1(x)$ and $\varepsilon_{j'} \cdot s_2(x)$. Then, U_i pretends U_v to compute the session key K_{j_2} using $\varepsilon_{j'} \cdot s_1(v), \varepsilon_{j'} \cdot s_2(v)$ and $M_{j_2}^{j'}(x), N_{j_2}^{j'}(x)$ from the broadcast message B_{j_2}.

Note that U_i is revoked in session j_2, thus he should not have computed K_{j_2}. Therefore the scheme cannot achieve the forward security. When the revoked user U_i obtains the session key K_{j_2}, he can of course give this session key to a new user who joins the group after session j_2 and should not know K_{j_2}. Hence, the scheme can not resist the collusion attack. Similarly, the scheme does not have the mt-revocation capability.

3.2. Analysis of the Weakness

Chen *et al.* [2] proposed two one-way hash chain self-healing group key distribution schemes based on the revocation polynomial in their paper. In fact, in the first scheme, each $k_j^{j'}$ is masked by different masking polynomials, $\{\varepsilon_{j'} \cdot s_j(x) | j = j', j'+1, \cdots, m\}$, which makes the scheme to be more secure. However, Chen *et al.* claimed that using multiple masking polynomials does not contribute to the security. Based on this consideration, they presented the second scheme only using one masking polynomial for each $k_j^{j'}$ to reduce the number of masking polynomials and the personal secret stored by each user. Thus the second scheme achieves the optimal storage overhead.

Now let us check the attack again. From the above attack, it is easy to find that only using one masking polynomial to construct the personal secret directly makes the Equation (6) (in step 4) hold, where $\varepsilon_{j'} \cdot s_1(x)$ disappears when Equation (4) minus Equation (5). Furthermore, $\varepsilon_{j'} \cdot s_2(x)$ can be computed by the revoked user U_i through the Equation (7), which leads to the exposure of those users' personal secret who join the group in session j', and finally results in the exposure of the session keys which should be kept secret from U_i.

Chen *et al.* [2] list Theorem 5 to show the security of their Scheme 2, thus Theorem 5 does not hold. To sum up, multiple masking polynomials should be adopted to design a secure self-healing group key distribution schemes using the polynomial secret sharing as the basic cryptographic technique. Unfortunately, multiple masking polynomials brings in the linear storage overhead. How to design a secure self-healing group key distribution schemes with constant storage overhead based on the polynomial secret sharing technique is still an open problem.

4. Conclusions

Chen *et al.* claimed that their self-healing group key distribution Scheme 2 achieves all basic security properties. Unfortunately, we found that Chen *et al.*'s Scheme 2 is insecure. Some security flaws are pointed out in this paper, *i.e.*, the Scheme 2 can not hold the forward security, mt-revocation capability and mt-wise collusion attack resistance capability.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61300172, 61170295), High Technology Research and Development Program of China (No. 2015AA016004), the

Fund of the State Key Laboratory of Software Development Environment (No. SKLSDE-2014ZX-14), and the Fundamental Research Funds for the Central Universities No. YWF-15-SYS-JSJXY-004).

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Staddon, J.; Miner, S.; Franklin, M.; Balfanz, D.; Malkin, M.; Dean, D. Self-healing key distribution with revocation. In Proceedings of the 2002 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 12–15 May 2002; pp. 241–257.
2. Chen, H.; Xie, L.; Wang, Q. Improved One-Way Hash Chain and Revocation Polynomial-Based Self-Healing Group Key Distribution Schemes in Resource-Constrained Wireless Networks. *Sensors* **2014**, *14*, 24358–24380.