On the security of an identity-based authenticated group key agreement protocol for imbalanced mobile networks

Haiyan Sun (wenzhong2520@gmail.com)

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: Recently, Islam and Biswas proposed a pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile networks. However, in this letter, we point out that this protocol cannot resist passive attack, and cannot provide forward secrecy for joining operation and backward secrecy for leaving operation.

Keywords: passive attack, forward secrecy, backward secrecy

1. Introduction

Recently, Isalam and Biswas [1] proposed a new group key agreement protocol for imbalanced mobile networks, called HB-12 protocol. They claimed that their protocol can satisfy requirements: contributiveness, message integrity, resilience against passive attack and forward/backward secrecy for joining/removing operation. In this letter, however, we find that the HB-12 protocol is complete insecure since it cannot provide resilience against passive attack, forward secrecy for joining operation and backward secrecy for leaving operation.

The remaining part of this letter is organized as follows. A review of the HB-12 protocol is given in section 2. Security analysis of the HB-12 protocol is provided in section 3. Finally, some conclusions are drawn in section 4.

2. Review of the IB-12 protocol

The IB-12 protocol considers the following three assumptions. Firstly, let $U = \{U_1, U_2, \cdots, U_{n-1}\}$ be the set of low-power mobile nodes and U_n be the powerful node (authentication server) of the network, however, each $U_i (1 \le i \le n)$ can execute the proposed protocol. Secondly, each group member at beginning must know the identity of other group members by some sort of other mechanism so that secure group is formed. Thirdly, the node U_n has the authorization of adding and removing the low-power nodes from the group. The IB-12 protocol consists of five phases: setup phase, key extraction phase, authenticated group key agreement phase, remove phase and join phase.

2.1 Setup phase

This algorithm takes a security parameter $k \in Z_q^*$ as input, and returns system parameters and a master key. Given k, PKG does as follows:

(a) Choose a k-bit prime q and determine the tuple $\{F_q, E/F_q, G, P\}$, where the point P is the generator of G.

(b) Choose the master key $x \in Z_q^*$ and compute the system public key $P_{pub} = xP$.

(c) Choose two cryptographic secure hash functions $H_0 : \{0,1\}^* \times G \to Z_q^*$ and $H_1 : \{0,1\}^* \to \{0,1\}^k$.

(d) Publish $\Omega = \{F_q; E/F_q; G; P; P_{pub}; H_0; H_1\}$ as system parameters and keep the master key x secret.

2.2 Key extraction phase

This algorithm takes master private key, a user's identifier, and system parameters as input, and returns the user's identity-based long-term private key. With this algorithm, for a user U_i with identifier ID_i, PKG does as follows:

(a) Choose a number $r_i \in Z_q^*$, compute $R_i = r_i P$ and $h_i = H_0(ID_i \| R_i)$.

(b) Compute $d_i = r_i + h_i x$.

U_i's private long-term key is (d_i, R_i) and is transmitted to U_i via an authenticated and secure channel. The public key of the user U_i is $P_i = R_i + H_0(ID_i \| R_i)$ and then he/she can validate his/her private/public key pair by checking whether the equation $P_i = R_i + H_0(ID_i \| R_i) = d_i P$ holds. The private key/public key pair is valid if the above equation holds and vice versa.

2.3 Authenticated group key agreement phase

Step 1 (Round 1) In this round, each low-power node $U_i (1 \le i \le n-1)$ picks a number $a_i \in Z_q^*$ and performs the following:

(a) Compute $T_i = a_i P_i$ and $S_i = d_i (H_1(ID_i \| T_i) + a_i)$.

(b) Send the message (ID_i, T_i, S_i, R_i) to the powerful node U_n.

Step 2 (Round 2). Upon receiving each message (ID_i, T_i, S_i, R_i) from each low-power node $U_i (1 \le i \le n-1)$, the powerful node U_n selects a number $a_n \in Z_q^*$ and executes the following operations:

(a) U_n verifies whether the equation $S_i P - H_1(ID_i \| T_i) P_i = ? T_i$ holds for $1 \le i \le n-1$, where $P_i = R_i + H_0(ID_i \| R_i) P_{pub}$. If it holds, U_n can ensure that $(ID_i, T_i, S_i, R_i)(1 \le i \le n-1)$ are sent by $U_i (1 \le i \le n-1)$ and each of them are authentic.

(b) U_n computes $T_n = a_n P_n$ and $T = \sum_{i=1}^{n-1} T_i$.

(c) U_n computes $Z_i = a_n d_n (T - T_i)(1 \le i \le n-1)$ for the low-power node $U_i (1 \le i \le n-1)$.

(d) U_n computes $ID = ID_1 \| ID_2 \| \cdots \| ID_n$, $Z = Z_1 \| Z_2 \| \cdots \| Z_{n-1}$ and $S_n = d_n (H_1(ID_n \| T_n \| Z) + a_n)$.

(e) U_n computes $K = a_n d_n T$ and the session key $SK = H_1(ID \| Z \| K)$.

(f) Then the powerful node U_n broadcast the message $(ID_n, T_n, Z_1, Z_2, \cdots, Z_{n-1}, S_n, R_n)$ to other low-power node $U_i (1 \le i \le n-1)$.

Step 3 (Authenticated group session key computation). After receiving the broadcast message

$(ID_n, T_n, Z_1, Z_2, \cdots, Z_{n-1}, S_n, R_n)$ from U_n, each low-power node $U_i (1 \le i \le n-1)$ computes $ID = ID_1 \| ID_2 \| \cdots \| ID_n, Z = Z_1 \| Z_2 \| \cdots \| Z_{n-1}, P_n = R_n + H_0(ID_n \| R_n) P_{pub}$ and then verifies whether the equation $S_n P - H_1(ID_n \| T_n \| Z) P_n = ? T_n$ holds. If it holds, each node $U_i (1 \le i \le n-1)$ can ensure that the message $(ID_n, T_n, Z_1, Z_2, \cdots, Z_{n-1}, S_n, R_n)$ is authenticated and is sent by the powerful node U_n. Then each low-power node $U_i (1 \le i \le n-1)$ computes the partial session key $K_i = a_i d_i T_n + Z_i (1 \le i \le n-1)$ and the contributory group session key $SK = H_1(ID \| Z \| K)$, where $K_1 = K_2 = \cdots = K_{n-1} = K$.

2.4 Remove phase

When a user or a set of users wish to leave the group, the remove phase occurs and in this case, either a new group key or the modification of the existing group is necessary for the protection of the group. In this paper, we proposed a modification of the existing group key in such a way that none of the leaving user can compute the subsequent group key generated. Suppose that a set of low-power mobile nodes $\overline{U} = \{U_{j+1}, U_{j+2}, \cdots, U_{n-1}\}$ wish to leave the group, the proposed protocol for implementing the remove phases is given below.

Step 1.

(a) Each $U_k (j+1 \le k \le n-1)$ informs U_n as they wants to leave the group.

(b) U_n then updates the group $U' = U \setminus \overline{U}$.

(c) U_n selects $a_n' \in Z_q^*$, computes $T_n' = a_n' P_n$ and $T' = T - \sum_{i=j+1}^{n-1} T_i$.

(d) U_n computes $Z_i' = a_n' d_n (T' - T_i) (1 \le i \le j)$ for the low-power node $U_i (1 \le i \le j)$.

(e) U_n computes $ID' = ID_1 \| ID_2 \| \cdots \| ID_j \| ID_n, Z' = Z_1' \| Z_2' \| \cdots \| Z_j'$ and $S_n' = d_n (H_1(ID_n \| T_n' \| Z') + a_n')$.

(f) U_n computes $K' = a_n' d_n T$ and the session key $SK = H_1(ID' \| Z' \| K')$.

(g) Then U_n broadcast the message $(ID_n, T_n', Z_1', Z_2', \cdots, Z_j', S_n', R_n)$ to other low-power node $U_i (1 \le i \le j)$.

Step 2. On receiving $(ID_n, T_n', Z_1', Z_2', \cdots, Z_j', S_n', R_n)$, each low-power node $U_i (1 \le i \le j)$ computes $ID' = ID_1 \| ID_2 \| \cdots \| ID_j \| ID_n, Z' = Z_1' \| Z_2' \| \cdots \| Z_j', P_n = R_n + H_0(ID_n \| R_n) P_{pub}$ and verifies whether the equation $S_n' P - H_1(ID_n \| T_n' \| Z') P_n = ? T_n'$ holds. If it holds, each node $U_i (1 \le i \le j)$ authenticates the message $(ID_n, T_n', Z_1', Z_2', \cdots, Z_j', S_n', R_n)$ and its sender U_n. Subsequently, each $U_i (1 \le i \le j)$ computes the partial session key $K_i' = a_i d_i T_n' + Z_i' = K' (1 \le i \le j)$ and the contributory group session key $SK' = H_1(ID' \| Z' \| K')$.

2.5 Join phase

This phase occurs when a new user or a set of new users want to join the existing group. In order to provide the fairness in the group key formation, the existing group key in this case should also be updated by including the contributions of the new members, however, it should be done in such a manner that none of new members can compute any of the previous group session keys. Suppose that a set of low-power mobile nodes $\widehat{U} = \{U_{n+1}, U_{n+2}, \cdots, U_m\}$ wish to

join existing group. The new contributory group session key in this phase for the mobile nodes $U'' = U \bigcup \widehat{U}$ with U_n will be computed as follows:

Step 1.

(a) Each members of $\widehat{U}$ send their identity to U_n.

(b) U_n then updates the group $U'' = U \bigcup \widehat{U}$.

(c) Each $U_k (n+1 \le k \le m)$ picks a number $a_k \in Z_q^*$, computes $T_k = a_k P_k$ and $S_k = d_k (H_1(ID_k \| T_k) + a_k)$, and sends (ID_k, T_k, S_k, R_k) to U_n.

Step 2. Upon receiving messages $(ID_k, T_k, S_k, R_k)(n+1 \le k \le m)$, U_n selects executes the following operations:

(a) Un verifies the integrity of each $(ID_k, T_k, S_k, R_k)(n+1 \le k \le m)$ as discussed earlier.

(b) U_n selects $a_n'' \in Z_q^*$, computes $T_n'' = a_n'' P_n$ and $T'' = T + \sum_{i=n+1}^{m} T_i$.

(c) U_n computes $Z_i'' = a_n'' d_n (T'' - T_i)$ for the low-power node $U_i (1 \le i \le m, i \ne n)$.

(d) U_n computes $ID'' = ID_1 \| ID_2 \| \cdots \| ID_m, Z'' = Z_1'' \| Z_2'' \| \cdots \| Z_{n-1}'' \| Z_{n+1}'' \cdots \| Z_m''$ and $S_n'' = d_n (H_1(ID_n \| T_n'' \| Z'') + a_n'')$.

(e) U_n computes $K'' = a_n'' d_n T''$ and the session key $SK'' = H_1(ID'' \| Z'' \| K'')$.

(f) Then U_n broadcast the message $(ID_n, T_n'', Z_1'', Z_2'', \cdots, Z_{n-1}'', Z_{n+1}'', \cdots, Z_m'', S_n'', R_n)$ to each $U_i (1 \le i \le m, i \ne n)$.

Step 3 After receiving $(ID_n, T_n'', Z_1'', Z_2'', \cdots, Z_{n-1}'', Z_{n+1}'', \cdots, Z_m'', S_n'', R_n)$, $U_i (1 \le i \le m, i \ne n)$ computes $ID'' = ID_1 \| ID_2 \| \cdots \| ID_m, Z'' = Z_1'' \| Z_2'' \| \cdots \| Z_{n-1}'' \| Z_{n+1}'' \cdots \| Z_m'', P_n = R_n + H_0(ID_n \| R_n) P_{pub}$ and then verifies the integrity of the received message as said earlier. If the integrity check satisfies, then each low-power node $U_i (1 \le i \le m, i \ne n)$ confirms that the received message is really sent by U_n. Thereafter, each $U_i (1 \le i \le m, i \ne n)$ computes $K_i'' = a_i d_i T_n'' + Z_i'' = K'' (1 \le i \le m, i \ne n)$ and the contributory group session key $SK'' = H_1(ID'' \| Z'' \| K'')$.

3. Weaknesses of the HB-12 protocol

In this section, we will show that the HB-12 protocol [1] cannot resist passive attack, and cannot provide forward secrecy for joining operation and backward secrecy for leaving operation.

3.1 Passive attack

In the following, we show that a passive adversary $\mathcal{A}$ can obtain the session key. Assume adversary $\mathcal{A}$ has eavesdropped the transferred messages $(ID_n, T_n, Z_1, Z_2, \cdots, Z_{n-1}, S_n, R_n)$ and $\{(ID_i, T_i, S_i, R_i)\}_{1 \le i \le n-1}$.

(a) $\mathcal{A}$ computes $ID = ID_1 \| ID_2 \| \cdots \| ID_n$, $Z = Z_1 \| Z_2 \| \cdots \| Z_{n-1}$ and $T = \sum_{i=1}^{n-1} T_i$.

(b) From the values $(Z_1, Z_2, \cdots, Z_{n-1})$, $\mathcal{A}$ computes $Y = \sum_{i=1}^{n-1} Z_i$ and $K = \frac{Y}{n-2}$.

(c) Then $\mathcal{A}$ can compute the session key as $SK = H_1(ID \| Z \| K)$.

Thus the HB-12 protocol cannot resist passive attack.

3.2 Failure to provide forward secrecy for joining operation

Suppose that a set of low-power mobile nodes $\widehat{U}=\{U_{n+1}, U_{n+2},\cdots,U_m\}$ wish to join existing group. In the following, we show that any $U_k(n+1\le k\le m)$ can generate any previously established session key between $U_i(1\le i\le n)$.

(a) $U_k(n+1\le k\le m)$ eavesdropped the transferred messages $(ID_n,T_n,Z_1,Z_2,\cdots,Z_{n-1},S_n,R_n)$ and $\{(ID_i,T_i,S_i,R_i)\}_{1\le i\le n-1}$.

(b) $U_k(n+1\le k\le m)$ computes $ID=ID_1||ID_2||\cdots||ID_n$, $Z=Z_1||Z_2||\cdots||Z_{n-1}$ and $T=\sum_{i=1}^{n-1}T_i$.

(c) From the values $(Z_1,Z_2,\cdots,Z_{n-1})$, $U_k(n+1\le k\le m)$ computes $Y=\sum_{i=1}^{n-1}Z_i$ and $K=\frac{Y}{n-2}$.

(d) Then $U_k(n+1\le k\le m)$ can compute the session key as $SK=H_1(ID||Z||K)$.

Thus the HB-12 protocol cannot provide forward secrecy for joining operation.

3.3 Failure to provide backward secrecy for leaving operation

Suppose that a set of low-power mobile nodes $\overline{U}=\{U_{j+1}, U_{j+2},\cdots,U_{n-1}\}$ wish to leave the group. In the following, we show that any $U_k(j+1\le k\le n-1)$ can generate any subsequent session key between $U_i(i\in[1,j]\bigcup\{n\})$.

(a) $U_k(j+1\le k\le n-1)$ eavesdropped the transferred messages $(ID_n,T_n,Z_1,Z_2,\cdots,Z_j,S_n,R_n)$ and $\{(ID_i,T_i,S_i,R_i)\}_{1\le i\le j}$.

(b) $U_k(n+1\le k\le m)$ computes $ID=ID_1||ID_2||\cdots||ID_j||ID_n$, $Z=Z_1||Z_2||\cdots||Z_j$ and $T=\sum_{i=1}^{j}T_i$.

(c) From the values $(Z_1,Z_2,\cdots,Z_j)$, $U_k(j+1\le k\le n-1)$ computes $Y=\sum_{i=1}^{j}Z_i$ and $K=\frac{Y}{j-1}$.

(d) Then $U_k(n+1\le k\le m)$ can compute the session key as $SK=H_1(ID||Z||K)$.

Thus the HB-12 protocol cannot provide backward secrecy for leaving operation.

4. Conclusion

In this letter, we have pointed out that Islam and Biswas's protocol cannot resist passive attack, and cannot provide forward secrecy for joining operation and backward secrecy for leaving operation.

References

[1] S. H. Islam and G. P. Biswas, "A pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile networks," Annals of Telecommunications, vol. 67, no. 11-12, 2012, pp. 547-558.