Compromise-Resistant Pairwise Key Establishments for Mobile Ad hoc Networks

Gicheol Wang and Gihwan Cho

ABSTRACT—This letter presents a pairwise key establishment scheme that is robust against the compromise of nodes in mobile ad hoc networks. Each node establishes local keys with its neighbor nodes that are at most three hops away at network boot-up time. When any two nodes establish a pairwise key, they receive the secret information from the nodes on the route between them, and construct the pairwise key using the secret information. Here, the local keys are utilized by the nodes on the route to send the secret information securely. The simulation results have proven that the proposed scheme provides better security than the key pre-distribution-based scheme.

Keywords—Pairwise key establishment, mobile ad hoc network, key management.

I. Introduction

Pairwise key establishment between any two nodes is more important in ad hoc networks than in wired networks. This is because all communications in ad hoc networks are carried out via an unreliable air medium.

In most of the pairwise key establishment schemes for ad hoc networks, each node obtains a set of keys from a key server prior to joining a network and establishes a pairwise key with other nodes using these keys [1]-[3]. However, it induces a significant amount of communication overhead to search a required number of proxies. More importantly, as the number of nodes compromised by attackers increases, the security of the pairwise key deteriorates significantly. This is because many keys within a node also exist within other nodes in a network with a predefined probability.

In other schemes, each node establishes one-hop pairwise keys with its neighbors using a shared key among all nodes, and establishes further hop pairwise keys using these one-hop pairwise keys [4], [5]. However, this approach can only be applied to a static network such as a sensor network. If this scheme is applied to a mobile network, each node is unable to find most of neighbors with which it established one-hop pairwise keys due to the node mobility. As a result, most trials for key establishment result in failure.

In the proposed scheme, instead of obtaining some keys from a server, each node establishes local keys with its neighboring nodes which are at most d (>1) hops away at network boot-up time, and saves them in its memory. Each node considers the nodes sharing the same local key as friends and the other nodes as unknown nodes. After establishing local keys, each node gets on a move to any direction in the network. So, the friends of a node are widely distributed throughout the network. When any two nodes want to establish a pairwise key, the nodes on the route between them send their secret information to both nodes. Two nodes construct the pairwise key by performing an 'exclusive or' (XOR) operation with the secret information. Here, the nodes on the route employ the local keys to protect the secret information. However, if the exchange of secret information should take place between any unknown nodes, only their friends will assist the exchange by giving their local keys to them. So, it is important to look for a lot of friends during the exchange. If each node searches for its friends in a large range (for example, r (>2) hops), they may acquire a sufficient number of friends. However, this causes a significant communication overhead. Conversely, if each node established local keys with more neighbors (that is, a large value of d) at network boot-up time, there will be no need for each node to search for its friends in a large range.

Manuscript received Oct. 29, 2005; revised Jan. 04, 2006.

Gicheol Wang (phone:+82 63 270 3437, email: gcwang@dcs.chonbuk.ac.kr) is with the Center for Advanced Image and Information Technology, Chonbuk National University, Jeonju, Jeonbuk, Korea.

Gihwan Cho (email: ghcho@dcs.chonbuk.ac.kr) is with CAIIT, Division of Electronics and Information Engineering, Chonbuk National University, Jeonju, Jeonbuk, Korea

To minimize the overhead resulting from the search of friends, after assuming the r value to be 1, we determined a reasonable value of d through simulation. Each node measured how many local keys the neighbor friends share with unknown nodes. This measurement indicates the rate at which a node can acquire local keys shared with unknown nodes from its neighbor friends during the exchange of secret information.

As shown in Fig. 1, if each node has established local keys with its three-hop neighbors initially (that is, $d = 3$), a one-hop search is sufficient to find a local key that can be employed for the protection of any secret information under a mobile environment. Therefore, we assume that each node has established three-hop local keys using a key pre-shared among all nodes, which is called an initial network-wide key.

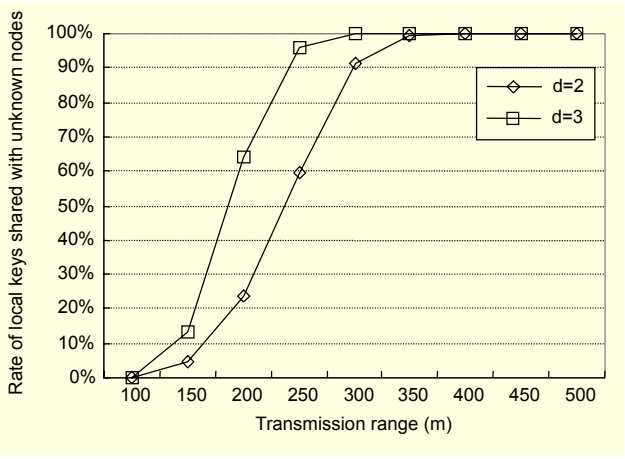

Fig. 1. Rate of local keys shared with unknown nodes vs. transmission range (100 nodes, 1000 m×1000 m, 16m/s).

II. Proposed Scheme

It is assumed that all nodes are initially trustworthy at network boot-up time, and an attacker cannot compromise a node during the initial three-hop local key establishment. This assumption is reasonable because the initial key establishment takes place prior to deploying the nodes into the workplace. Otherwise, an attacker acquiring the initial network-wide key can launch various attacks by disguising a legal user. After generating three-hop local keys, each node erases the initial network-wide key from its memory to prevent an attacker from joining a network. The following notations are employed in the rest of this letter.

– K_I : initial network-wide key
– F_K : pseudo-random function
– $\{M\}_K$: encryption of message M with a symmetric key K
– $MAC(K, M)$: message authentication code of M with a symmetric key K
– $nonce_A$: a random number generated by node A

First, each node establishes initial one-hop local keys with its neighbors through the following steps.

1. A node u generates a random number $nonce_u$ and broadcasts a hello message containing its identifier and a random number.
2. Node v receiving a hello message from a neighbor compares the neighbor's identifier with its identifier. If the neighbor's identifier is lower, it generates its master key, $K_v = F_{K_I}(v)$, and a one-hop local key using the neighbor's identifier and its master key, $K_{uv} = F_{K_v}(u)$. Then it computes the message authentication code (MAC) value for $nonce_u$ and its identifier using its master key, and responds to node u with the MAC value. If node v receives the hello message from a higher ID node, it does not respond.
3. Node u generates node v's master key, $K_v = F_{K_I}(v)$, and the one-hop local key shared with node v, $K_{uv} = F_{K_v}(u)$, after verifying node v's legality by computing the MAC value.

After establishing one-hop local keys, each node broadcasts a list of nodes with which it established one-hop local keys, and the list is encrypted with the initial network-wide key. Establishment of two-hop local keys is performed via the comparison of the identifier as step 2 in the one-hop local key establishment. For instance, node u receives a list of nodes from neighbor v (for example, $\{b, u, y\}$). First, node u generates master keys of node b and y, that is, $K_b = F_{K_I}(b)$, $K_y = F_{K_I}(y)$. Then, because node u's identifier is higher than node b's, it establishes the two-hop local key with node b using its master key, that is, $K_{bu} = F_{K_u}(b)$. In the case of node y, because u's identifier is lower than y's, it establishes the two-hop local key with node y using y's master key, that is, $K_{uy} = F_{K_y}(u)$. Three-hop local keys are established through the same manner as the two-hop local key establishment.

Now, let us assume that any two nodes (for example, u and v) want to establish a pairwise key as shown in Fig. 2. Here, the route between them is assumed to be established securely via a secure routing protocol.

Source u generates its secret key and tries to send it to destination v. Because node u has no local key shared with v and no neighboring friends, it discards its secret key and sends the key setup message to the next node w.

Node w generates its secret key (k_w) and tries to send it to u and v. Because w has no local key shared with v, it obtains a local key from its neighboring friends. Node w obtains a local key, K_{jv}, from a neighboring friend j. Then, node w sends its secret key to u using K_{uw} and to v using K_{jv}. Next, node w sends the key setup message to the next node z.

Node z should obtain two local keys from its neighboring friends because it shares no local keys with u and v. It obtains

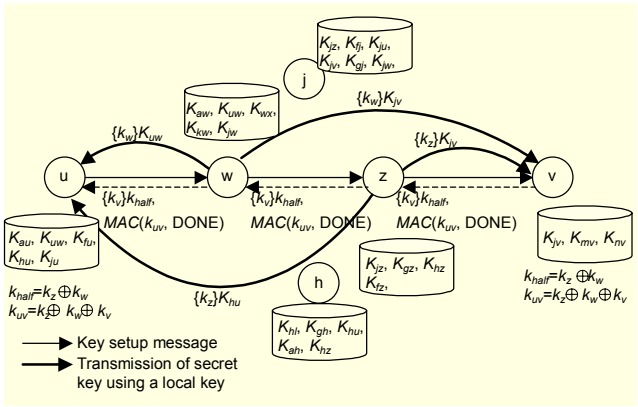

Fig. 2. Procedure of pairwise key establishment.

Table 1. Simulation parameters and their values.

Parameter	Value
Routing protocol	Dynamic Source Routing [6]
Simulation time	180 seconds
Period of key establishment	2 seconds
Maximum connections of node pair trying key establishment	30
Exposure probability of pairwise key	0.000001 (proven to be reasonable in [7])
Size of key pool	2000 keys
Number of pre-distributed keys	160 keys

two local keys from j, that is, K_{ju} and K_{jv}, and one local key from h , K_{hu}. Node z sends its secret key (for example, k_z) to u using K_{hu} and to v using K_{jv}. Then, node z sends the key setup message to the next node v.

Destination v generates a key for encryption of its secret key by performing an XOR operation with the received secret keys, that is, k_w and k_z. Also, v constructs the pairwise key shared with the source u by performing an XOR operation with its secret key and received secret keys. Then v generates a response message and sends it to source u. The response message consists of the encrypted secret key of v and the *MAC* value of a DONE message authenticated with the pairwise key.

After decrypting the destination's secret key, source u constructs the pairwise key shared with destination v by performing an XOR operation with the v's decrypted secret key and the pre-received secret keys. Lastly, u verifies the correctness of the pairwise key by confirming the DONE message.

In the proposed scheme, if the destination receives at least one secret key from a node on the route, the key establishment succeeds. Otherwise, the key establishment falls into failure, and the source retries the establishment of the pairwise key at a later time.

When a new node joins a network, it establishes one-hop local keys with the existing nodes having higher identifiers. Contrarily, the existing nodes having lower identifiers cannot establish one-hop local keys with the new node because they need the initial network-wide key to generate the master key of the new node. An existing node that has been reset cannot rejoin the network unless it obtains the initial network-wide key from the administration server.

III. Security Analyses

To analyze the security of the proposed scheme, simulations have been conducted using the ns-2 network simulator. The proposed scheme was compared to the key pre-distribution-based scheme and its extended version, which employs two-hop proxies [3]. The primary difference of both versions is the search range of proxy nodes. That is, in the original version, the source node searches its proxy nodes in its one-hop range, while the extended version increases the search range to two hops. Table 1 presents the parameters and their values used in the simulations.

Through the simulations, the extent to which the number of compromised nodes affected the security of the pairwise key was analyzed. To this end, the exposure rate of the initial local keys or pre-distributed keys employed during a pairwise key establishment was measured as the number of compromised nodes increased. Also, when the compromised nodes were limited to a specific number, the rate was measured as the number of nodes increased.

As shown in Fig. 3(a), both schemes increase the exposure rate as the number of compromised nodes increases. In the key pre-distribution-based scheme using one-hop proxy nodes, even though the rate of compromised nodes is low, 30%, an attacker can obtain 80% of pre-distributed keys. Furthermore, if the rate of compromised nodes reaches 60%, all pre-distributed keys are exposed to attackers. Even in the extended version, 80% of pre-distributed keys employed during a pairwise key establishment are exposed to an attacker when the rate of compromised nodes reaches 60%.

In contrast, in the proposed scheme, even though 60% of nodes are compromised, attackers can acquire only 20% of initial local keys that are employed during a pairwise key establishment. This is because the number of initial local keys is much smaller, and the utilization frequency of the initial local keys is much lower than that of the pre-distributed keys.

Also, even if an attacker gets a pairwise key established between any two nodes at the current time, it can hardly get a pairwise key established between the same nodes the next time. This is because the route between the same nodes may change

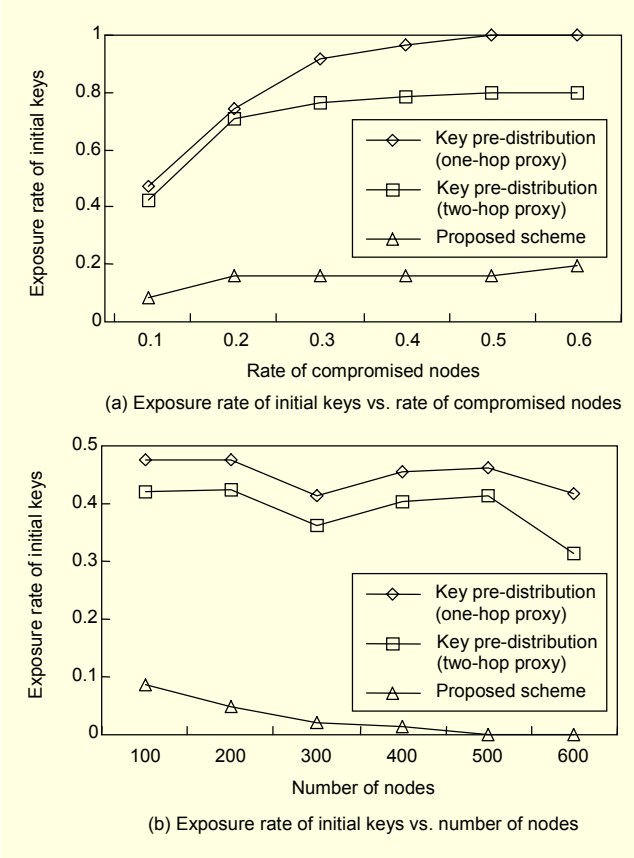

Fig. 3. Exposure rate of initial keys (mobility : 8 m/s, simulation area : 1000 m ×1000 m).

with the lapse of time.

Next, after setting the compromised nodes to 10, the exposure rate of initial local keys or pre-distributed keys employed for a key establishment was measured as the number of nodes increased. As shown in Fig. 3(b), the key pre-distribution-based scheme is not greatly affected by the increase of nodes. In other words, it is greatly affected by even a small number of compromised nodes. Conversely, in the proposed scheme, the rate decreases as the number of nodes increases. This indicates that this approach provides better security to the network if the network can restrict the number of compromised nodes to a specific number.

IV. Conclusions

The proposed scheme requires no pre-distribution of keys and forces each node to generate initial local keys within at most three hops so as to employ them for later establishment of pairwise keys. Because the number of initial local keys and their utilization frequency is limited, the increase of compromised nodes cannot have a great effect on the security of pairwise key establishment. The simulation results revealed that the proposed scheme minimizes the threat, which is caused by the compromised nodes, even if the number of compromised nodes increases.

References

[1] L. Eschenauer and V. D. Gilgor, "A Key-Management Scheme for Distributed Sensor Networks," *Proc. 9th ACM CCS '02,* 2002, pp. 41-47.

[2] W. Du J. Deng, Y. Han, S. Chen, and P. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," *Proc. IEEE INFOCOM 2004*, vol. 1, 2004, pp. 586-597.

[3] S. Zhu S. Xu, S. Setia, and S. Jajodia, "Establishing Pairwise Keys For Secure Communication in Ad Hoc Networks: A Probabilistic Approach," *Proc. IEEE 11th ICNP 2003*, 2003, pp. 326-335.

[4] S. Zhu, S. Setia, and S. Jajodia, "LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks," *Proc. 10th ACM CCS '03*, 2003, pp. 62-72.

[5] B. Dutertre, S. Cheung, and J. Levy, "Lightweight Key Management in Wireless Sensor Networks by Leveraging Initial Trust," *SRI Int'l Tech. Rep.*, SRI-SDL-04-02, Apr. 6, 2004.

[6] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing for Multihop Wireless Ad Hoc Networks," *Ad Hoc Networking*, *Ed.* C. E. Perkins, Addison-Wesley, 2000, pp. 139-168.

[7] R. Canetti et al., "Multicast Security: A Taxonomy and Some Efficient Constructions," *Proc. IEEE INFOCOM 1999*, vol. 2, 1999, pp. 708-716.