Group Key Based Session Key Establishment Protocol for a Secure Remote Vehicle Diagnosis

Sarang Wi, Jong-Hyouk Lee

Protocol Engineering Lab, Sangmyung University, Republic of Korea
{sarang, jonghyouk}@pel.smuc.ac.kr

Abstract

In this paper, a group key based session key establishment protocol is introduced for a secure remote vehicle diagnosis. The proposed scheme aims at providing secure authentication and session key establishment between a vehicle manufacturer server and a group of in-vehicle electronic control unit based on a key graph. The detailed operations of the proposed scheme are provided while providing security analysis results of the proposal. Thanks to the use of the group key, the proposed scheme reduces the number of keys and management cost for the secure remote vehicle diagnosis compared with a traditional scheme that uses a symmetric key cryptosystem.

Keywords: Key graph, Group key, Session key

1 Introduction

As advanced wireless communication and networking technologies are continuously developed, technologies for vehicular communication are emerging. Connected cars are expected to be in our daily life soon. Already various vehicular communication-based services such as traffic information notification, location based services, and vehicle remote diagnostic services have been studied and will be available when the Intelligent Transport Systems (ITS) is deployed [1]. Various research for vehicular communication are ongoing [2-8].

A connected car is considered as a computing device that moves along roads. This is also considered as a part of the Internet of Things (IoT), i.e., connected cars will be connected to the Internet as key players for the IoT. BI Intelligence expects that the connected car will occupy 75% of the world automotive production in 2020 [9]. This computing device will be not only providing web browsing and video streaming for users, but also used as a vehicle that carries people inside. As like personal computers and smartphones, the connected car is connected to other cars and to the Internet so that cyber threats will target the connected car as well. There have been many reports that current connected cars are vulnerable again various security attacks and many security experts successfully showed possible attack scenarios. For instance, an attacker can obtain vehicle state information and use this information maliciously to control the vehicle speed and breaking system [10]. Also, there are high privacy concerns related to vehicular communication [11-14].

As a preliminary work, we studied a session key establishment protocol for a vehicle diagnostic that was based on symmetric key cryptosystem [15]. However, the previously proposed protocol has practical issues, e.g., the number of symmetric key increases as the number of the Electronic Control Units (ECUs) in a car increases. In other words, in terms of key management, the preliminary work is inefficient due to the required number of symmetric keys. To address this issue, in this paper, we are focused on developing a secure session key establishment protocol for a vehicle diagnosis based on group key graphs. This paper is an extended version of the paper published in the Proc. of Qshine 2016 [16].

The rest of this paper is as follows. Section 2 presents the related work. Section 3 presents the proposed scheme. Security analysis results of proposed scheme are provided in Section 4. Section 5 concludes this paper.

2 Related Work

2.1 Vehicular Network Technologies

Vehicular networks can be divided into an internal vehicle network and an external vehicle network. The external vehicle network is used for vehicle-to-vehicle (V2V) communication and vehicle-to-infrastructure (V2I) communication, while the internal vehicle network is used for sharing data between ECUs in the car.

While new network technologies are being developed, existing network technologies such as Controller Area Network (CAN), Local Interconnect Network (LIN), Media Oriented Systems Transport

*Corresponding Author: Jong-Hyouk Lee; E-mail: jonghyouk@pel.smuc.ac.kr
DOI: 10.3966/160792642018051903001

(MOST), and Flexray are still major network technologies for the internal vehicle network [17-18]. For the V2V and V2I communications, relative new technologies such as IEEE 802.11p, LTE, IPv6, etc. are being deployed.

An in-vehicle gateway (i.e., vehicular mobile router located at a car) is used to connect the internal vehicle network and the external vehicle network. Data packets destined for Internet based services are sent from nodes (e.g., ECUs) on the internal vehicle network via the in-vehicle gateway. Similarly, data packets destined for the nodes on the internal vehicle network are sent from Internet based services to the nodes via the in-vehicle gateway.

For various services for connected cars, the end-to-end communication between a node located at the external vehicle network (e.g., at the Internet) and a node located at the internal vehicle network (e.g., at the CAN). Previously proposed security protections for vehicles have not much considered this issue so that security designs considering mainly separated network protections have been deployed. However, as the need for various remote management (e.g., the remote vehicle diagnosis) is highly required in connected car scenarios, a means for securing the end-to-end communication is required.

2.2 Group Key

There will be many useful use cases that require transmitting secure data to members of a group. However, if each member, e.g., ECU, on the internal vehicle network, uses a pair of symmetric keys to encrypt the data, it would be inefficient in terms of network bandwidth, computation, and key management cost. To overcome this limitation, a group key has been introduced that all members of a group share a same key. The important part of a group key is a group dynamic, which means that the members of the group can be changed. The main requirements for group keying are defined as follows [19].

Forward secrecy. when a member leaves from the group, the member who knows the old group key should not be able to know a new group key.

Backward secrecy. when a member joins to the group, the member who knows the current group key should not be able to know the old group key.

2.3 Key Graphs As a Group Key Management Model

For a group key management model, we adopt the key graphs introduced in [20-21].

Secure group. A secure group is *(E, K, R)* where *E* is a set of ECUs, *K* is a set of keys, and *R* denotes $R \subset E \times K$, which is a ECU-key relation. A key is held by each ECU in *E*.

We assume that a secure key management server (e.g., operated by a vehicle manufacturer) manages the group keys. The server securely distributes a key to a member of the ECU group and maintains a relation *R* between the ECU and the key of the group. Each ECU of the group has a set of keys: *individual key* of ECU, a *sub-group key*, and a *group key*. The *individual key* is shared only with the key management server. Let *z* is the name of a group. For *z*, the group key is K_{Gr_z}, which is used to send a message securely to other ECUs belonging to *z*.

Key graph. A key graph has two types: *e-node* representing ECUs and *k-node* representing keys. Each *e-node* has an outgoing edge but no incoming edge. Each *k-node* has an incoming edge. Among *k-node*, the top of *k-node* is called as a root node. The root node is single. The key graph has two parameters: *height* and *degree*. The *height* is the distance between the root node and uttermost with the root node and end node. The *degree* is the maximum number of incoming edges in the tree.

Relation of a key graph and a secure group. A relation of a key graph *G* and a secure group (*E, K, R*) is as follows:

(1) *E* and the set of *e-node* is a one-to-one correspondence in *G*.

(2) *K* and the set of *k-node* is a one-to-one correspondence in *G*.

(3) *R* constitutes (*e, k*).

Note that *G* shows a directed path between *e-node* that corresponds *e* and *k-node* that corresponds to *k*.

As an example of key management, let assume that there exists nine ECUs. The ECUs are divided into three subgroups. The subgroups are $\{e_1, e_2, e_3\}$, $\{e_4, e_5, e_6\}$ and $\{e_7, e_8, e_9\}$. Each ECU has three keys: *individual key*, entire *group key*, and *sub-group key*. In Figure 1, the tree key graphs *G* specifies the secure group (*E, K, R*).

$$
\begin{aligned}
E &= \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\} \\
K &= \{ke_1, ke_2, ke_3, ke_4, ke_5, ke_6, ke_7, ke_8, ke_9, \\
&\quad ke_{123}, ke_{456}, ke_{789}, ke_{1-9}\} \\
R &= \{(e_1, ke_1), (e_1, ke_{123}), (e_1, ke_{1-9}) \\
&\quad (e_2, ke_2), (e_2, ke_{123}), (e_2, ke_{1-9}) \\
&\quad (e_3, ke_3), (e_3, ke_{123}), (e_3, ke_{1-9}) \\
&\quad (e_4, ke_4), (e_4, ke_{456}), (e_4, ke_{1-9}) \\
&\quad (e_5, ke_5), (e_5, ke_{456}), (e_5, ke_{1-9}) \\
&\quad (e_6, ke_6), (e_6, ke_{456}), (e_6, ke_{1-9}) \\
&\quad (e_7, ke_7), (e_7, ke_{789}), (e_7, ke_{1-9}) \\
&\quad (e_8, ke_8), (e_8, ke_{789}), (e_8, ke_{1-9}) \\
&\quad (e_9, ke_9), (e_9, ke_{789}), (e_9, ke_{1-9})\}
\end{aligned}
$$

Here, ke_{1-9} is a group key. The following two functions are defined for the secure group (*E, K, R*):

$$
\begin{aligned}
&\textit{Keyset}\ (\mathrm{e}) = \{\ k \mid (e, k) \in R\ \} \\
&\textit{ECUset}\ (\mathrm{k}) = \{\ e \mid (e, k) \in R\ \}
\end{aligned}
$$

Keyset(e) is a set of keys that has ECU *e* in *E*.

ECUset(k) is a set of ECUs that has key *k* in *K*. For example, applying this function in Figure 1, we have *keyset* $(e_3) = \{ke_3, ke_{123}, ke_{1-9}\}$ and *ECUset* $(ke_{456}) = \{e_4, e_5, e_6\}$.

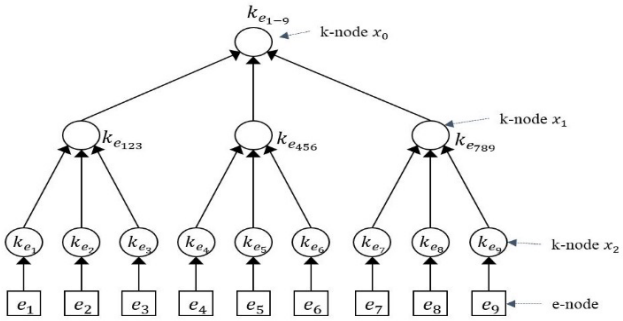

Figure 1. Tree key graph

3 Proposed Scheme

We present the proposed scheme designed for establishing a secure session key between a vehicle manufacturer server and an ECU of a vehicle based on a group key. The session key is then used for instance to encrypt and decrypt data communications between the server and the ECU for a secure remote vehicle diagnosis.

3.1 Notation and Assumption

Before explaining used notations and assumptions in this paper, we explain main entities of communications. The main entities are a server of vehicle manufacturer *S*, in-vehicle gateway *Gw*, and vehicle ECU E_i. Table 1 shows the notations used for the proposed scheme.

Table 1. Notation

Notation	Definition
S	vehicle manufacturer's key management server
Gw	in-vehicle gateway
E_i	vehicle ECU
Gr_z	vehicle ECU group *z*
ID_s	*ID* of *S*
ID_{GW}	*ID* of *Gw*
ID_{E_i}	*ID* of E_i
ID_{Gr_z}	*ID* of Gr_z
J	secret key of *Gw*
K	shared key between *S* and *Gw*
Q	secret key of *Gw*; used for creating a group key
k_{e_i}	secret key between *Gw* and E_i
K_{Gr_z}	group key of ECU group z
$SK_s - Gr_z$	session key between *S* and Gr_z
$E_{key}[]$	symmetric encryption
$D_{key}[]$	symmetric decryption
r	server's nonce
h(.)	one-way cryptographic hash function

The assumptions are as followings.

(1) When a vehicle is shipped, *S* is authenticated with *Gw*. A secure channel between *S* and *Gw* is established.

(2) When the vehicle is shipped, *Gw* has a three-secure key *J, K, Q*.

(3) When the vehicle is shipped, *Gw* and E_i share a secure key k_{e_i}.

3.2 Operation Process

Vehicle registration with a server. When a vehicle is produced, *Gw* is registered with *S*. In this time, *S* sends *ID* to *Gw* securely. *Gw* computes $C_{ig} = E_j[ID_s \| ID_{GW}]$. Only *Gw* creates C_{ig}. Here, C_{ig} is used for mutual authentication between *S* and *Gw*. *Gw* sends C_{ig} and *K* to *S* through a secure channel. Note that *J* and *K* are secure keys in *Gw* for a long time.

Authentication between a vehicle and a server. For communications between *S* and E_i, *Gw* makes a group key of each ECU group and sends a message including the group key, which is encrypted by each individual key of E_i. E_i decrypts the message using the own individual key. It is possible that it does encrypted communication between *Gw* and E_i. Figure 2 shows the session key establishment process between vehicle and *S*.

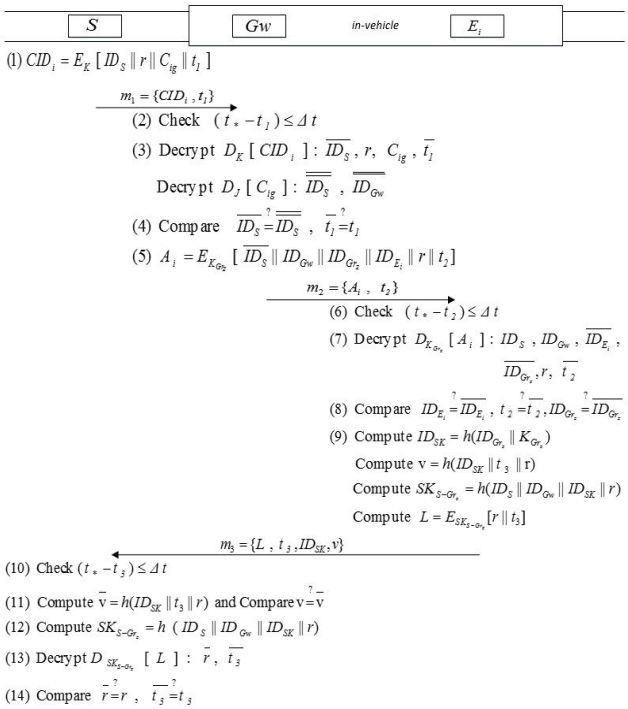

Figure 2. Session key establishment

(1) *S* computes $CID_i = E_K[ID_s \| r \| C_{ig} \| t_1]$, where ID_s is *ID* of server, *r* is a random number, C_{ig} is received from *Gw* when the vehicle is manufactured, and t_1 is the current time of *S*. Thereafter, *S* sends a

message $m_1 = \{CID_i, t_1\}$ to *Gw*. Note that CID_i is used for authentication between *S* and *Gw*.

(2) Upon receiving the message, *Gw* performs the following steps. Check, if $(t_* - t_1) \le \Delta t$, where Δt is the time interval for the transmission delay and t_* is the current time of *Gw*. If yes, then *Gw* performs the next process. Otherwise, it rejects the request and aborts any further process.

(3) *Gw* decrypts the message CID_i, using key *K* (i.e., $D_K[CID_i]$) and obtains $\overline{ID_s}, r, C_{ig}$ and $\overline{t_i}$. Similarly, it decrypts the message C_{ig} using key *J* (i.e., $D_J[C_{ig}]$) and obtains $\overline{ID_s}$ and $\overline{\overline{ID_{GW}}}$.

(4) *Gw* compares $\overline{ID_s} = \overline{\overline{ID_s}}$ and $t_1 = \overline{t_1}$. If yes, then *Gw* continues with the next steps; otherwise it aborts the request.

(5) *Gw* computes $A_i = E_{K_{Gr_z}}[\overline{ID_s} \| ID_{GW} \| ID_{Gr_z} \| ID_{E_i} \| r \| t_2]$, where t_2 is the current time of *Gw*. Thereafter, *Gw* sends the message $m_2 = \{A_i, t_2\}$ to E_i. A_i is used for authentication between *Gw* and E_i. This process delivers *ID* of *S*, *r* to create the session key of E_i.

(6) Upon receiving the message from *Gw*, E_i performs the following steps. Check, if $(t_* - t_2) \le \Delta t$, where t_* is the current time of E_i. If yes, then E_i performs the next process. Otherwise, it rejects the request and aborts any further process.

(7) E_i decrypts the message A_i using group key K_{Gr_z} (*i.e.*, $D_{KGr_z}[A_i]$) and obtains $ID_s, ID_{GW}, \overline{ID_{E_1}}, \overline{ID_{Gr_z}}, r, \overline{t_2}$.

(8) E_i compares $ID_{E_i} = \overline{ID_{E_i}}, t_2 = \overline{t_2}, ID_{Gr_z} = \overline{ID_{Gr_z}}$. If yes, then E_i continues with the next steps; otherwise it aborts the request.

(9) E_i computes $ID_{SK} = h(ID_{Gr_z} \| K_{Gr_z})$, $v = h(ID_{SK} \| t_3 \| r)$, session key $SK_{S-Gr_z} = h(ID_S \| ID_{GW} \| ID_{SK} \|)$, and $L = E_{SK_{S-Gr_z}}[r \| t_3]$, where t_3 is the current time of E_i. After that, E_i sends the message $m_3 = \{L, ID_{SK}, v, t_3\}$ to S.

(10) Upon receiving the message from E_i, *S* validates the time as follows. Check, if $(t_* - t_3) \le \Delta t$, where t_* is the current time of *S*. If yes, *S* performs the next process. Otherwise, *S* rejects the request and aborts any further process.

(11) *S* computes $\overline{v} = h(ID_{SK} \| t_3 \| r)$ and compares $v = \overline{v}$. If yes, then *S* continues with the next steps; otherwise it aborts the request.

(12) *S* computes session key $SK_{S-Gr_z} = h(ID_S \| ID_{GW} \| ID_{SK} \| r)$.

(13) *S* decrypts the message *L* using key SK_{S-Gr_z} (i.e., $D_{SK_{s-cr_z}}[L]$) and obtains $\overline{r}, \overline{t_3}$.

(14) S compares $r = \overline{r}, t_3 = \overline{t_3}$. If yes, then a secure session key is established between *S* and E_i; otherwise not.

When the session key establishment is completed, E_i sends a message including the session key to other ECUs in the same group so that other ECUs will have the session key for secure communications with *S*.

3.3 Key Management among the ECUs

In this section, we present the key management for the ECUs based on a group key. We concentrate upon the cases of a group joining and group leaving. The group key cryptosystem should create new keys (i.e., new individual key, new subgroup key, new group key) for the joining and leaving events. Note that the forward security and the backward security must be supported when a new user joins and a user exists.

Joining and leaving of a group happen in two occasions. The first occurs when a registration happens, e.g., all ECUs are registered to the server when the vehicle is shipped. The second occurs when a new ECU is installed, e.g., when an old ECU is replaced with a new one in a garage.

Joining. An ECU e_u which wants to join the secure group sends a requesting message to join to the key distribution server. This key server manages a group and has access management authority. When the server receives the requesting message, it begins an exchange for authentication ECU e_u. If a join request is approved, the server and ECU e_u have the key k_{e_u}.

The server creates a new *e-node* of ECU e_u and new *k-node* of e_u's *individual key* to change the tree key graphs. The server looks for joining point which we call parent node to attach newly created *k-node* in tree key graphs, and attaches this *k-node*. After the server creates a *new group key*. To prevent a join of new member e_u in the past communication access, the key must be changed from the joining point to the root node. These keys must be delivered safely to a joining member and existing members. The key management server thus encrypts these keys by *previous group key* for the existing members and *individual key* for the joining member.

If the server authorizes the ECU e_u and distributes the key k_{e_u} to e_u, that thing as follows. The server finds a joining point and attachs k_{e_u}. At the time x_j denotes the joining point, x_0 the root, and when $i = 1, \ldots, j$, x_{i-1} the parent of x_i. K_{j+1} denotes k_{e_u} and $K_0, \ldots, K_j$ the old keys of $x_0, \ldots, x_j$. The server

generates new keys $K_0', \ldots, K_j'$. The server sends $K_0', \ldots, K_j'$ to each ECU. When expressed as a function, it is $ECUset(K_0): E_{K_0}[K_0'], \ldots, E_{K_j}[K_j']$ And it sends $E_{k_{e_u}}[K_0', \ldots, K_j']$ to ECU e_u .

In Figure 3, e_9 sends a requesting message to join. And it is assumed that the approval for e_9 the secure group. The server creates a *new group key* ke_{1-9} and a *new sub-group key* ke_{789} which call joining point node's key. Among the existing ECU $e_1, e_2, \cdots, e_6$ need to *new group key* (i.e., ke_{1-9}) and e_7, e_8 need to a *new group key*, a *new sub-group key* (i.e., ke_{789} , ke_{1-9}). The server sends securely rekey message to distribute the key. Rekey message is as follows.

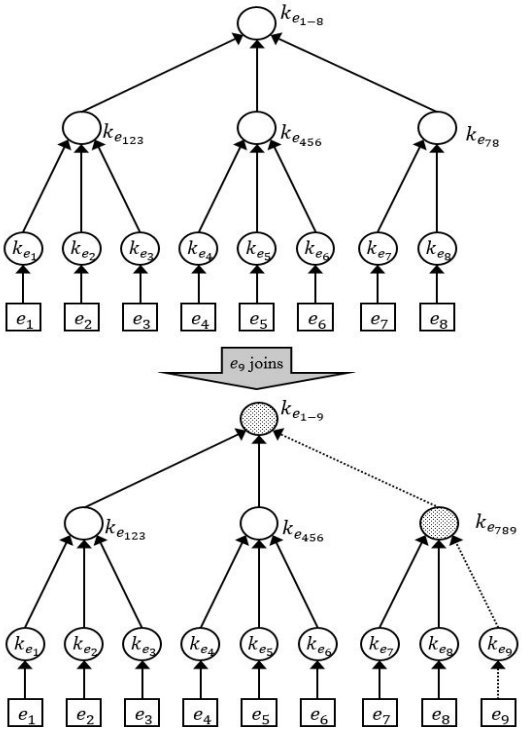

Figure 3. ECU e_9 requests to join in a tree key graphs

$$s \rightarrow \{e_1, e_2, \ldots, e_8\}: E_{ke_{1-8}}[k_{e_{1-9}}], E_{ke_{78}}[k_{e_{789}}] \quad \textbf{(1)}$$

$$s \rightarrow e_9 \quad : E_{ke_{89}}[k_{e_{1-9}}, k_{e_{789}}] \quad \textbf{(2)}$$

The server sends (1) message for $e_1, e_2, \ldots, e_8$ to multicast and sends (2) message for e_9 to unicast. If each ECU receives the messages, they would have only necessary information and discard unnecessary information. In this consist of rekey message, since the number of rekey messages is minimized, the overhead of the server is reduced.

Leaving. An ECU e_u which wants to leave the secure group sends requesting message to leave to the key distribution server. If the join request is approved, the server changes the tree key graphs. The server deletes the ECU e_u and k-node of e_u 's individual key. This k-node's parent called leaving point. To prevent the access of the leaving member, keys must be changed from leaving point to root node. The server creates a new group key and distributes securely to the remaining members.

If the server responds message to leave, that thing as follows. The server searches for a parent node of leaving e_u 's individual key k_{e_u} which is called leaving point. And remove k_{e_u} from the tree. x_{j+1} denote the deleted k-node for k_{e_u} , x_j the leaving point, x_0 the root, and when $i = 1, \ldots, j$, x_{i-1} the parent of x_i . The server generates randomly keys $K_0', \ldots, K_j'$ as the new keys of $x_0, \ldots, x_j$. And when $i = 0, \ldots, j$, $J_1, \ldots, J_r$ denote key at the children of x_i in the new tree key graphs. The server encrypts K_i' to each children key that called L_i . This denotes $E_{j_1}[K_i'], \cdots, E_{j_r}[K_i']$. When expressed as a function, it is $ECUset(K_0')$: $L_0, \ldots, L_j$.

In Figure 4, e_9 sends requesting message to leave. And it is assumed that the approval for e9 the secure group. The server creates a new group key $k_{e_{1-8}}$ and a new sub-group key $k_{e_{78}}$ which called leaving point node's key. Among the existing ECU $e_1, e_2, \cdots, e_6$ need to a new group key (i.e., $k_{e_{1-8}}$) and e_7, e_8 need to a new group key, a new sub-group key (i.e., $k_{e_{78}}$, $k_{e_{1-8}}$). The server sends securely rekey message to distribute the new key. The rekey message is as follows.

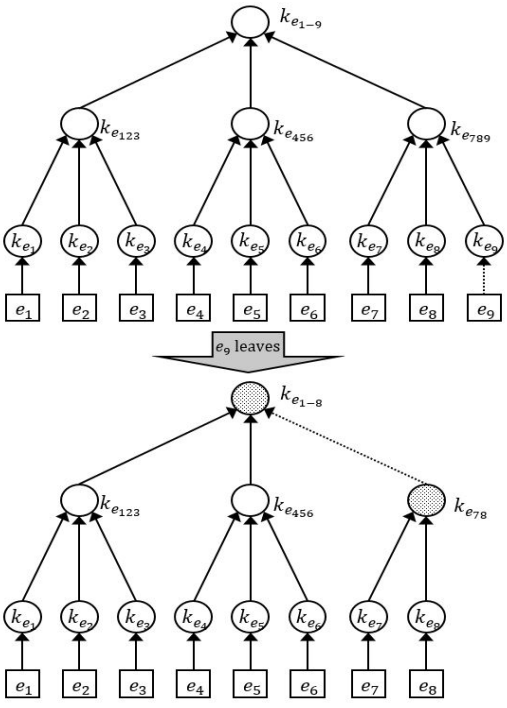

Figure 4. ECU e_9 requests to leave in a tree key graphs

Let L_0 denote

$$E_{k_{e_{123}}}[k_{e_{1-8}}], E_{k_{e_{456}}}[k_{e_{1-8}}], E_{k_{e_{789}}}[k_{e_{1-8}}]$$

Let L_1 denote

$$E_{k_{e_7}}[k_{e_{78}}], E_{k_{e_8}}[k_{e_{78}}] \rightarrow \{e_1, e_2, \ldots, e_8\} : L_0, L_1 \quad \textbf{(3)}$$

This approach uses only one rekey message. The server sends (3) message for $e_1, e_2, \cdots, e_8$ to multicast. A rekey message is configured to include all keys.

4 Security Analysis

In this section, we carry a security analysis to confirm that the proposed scheme provides confidentiality, mutual authentication, and secure session key establishment. In addition, it is shown that the proposed protocol could withstand a replay attack, assumed server attack, assumed gateway attack, and gateway secret assume attack. Note that the proposed scheme does not consider a system level security threats, e.g., the presence of malware on ECUs.

Theorem 1. The proposed security protocol could provide confidentiality.

Proof 1. To determine confidentiality of a message, it should be identified whether a malicious user can obtain a key for providing confidentiality or not. We examine that K, K_{Gr_z} , SK_{S-Gr_z} used in the proposed scheme. When a car is manufactured, the key K is shared between the vehicle gateway and the manufacturer server through a secure channel. The key K_{Gr_z} is delivered among a secure group, i.e., the gateway and a group of vehicle ECU. The key SK_{S-Gr_z} is generated as (i.e., $SK_{S-Gr_z} = h(ID_S \| ID_{GW} \| ID_{SK} \| r)$). Here, the parameters of making the session key cannot be obtained by malicious users. Thus, the proposed security protocol could provide confidentiality.

Theorem 2. The proposed security protocol could provide mutual authentication.

Proof 2. A communication between the server and the gateway become mutual authentication because used secure channel. And the remains of the two-other mutual authentication (i.e., the gateway-the ECU group, the ECU group-the server) explain through shown in Figure 1. The gateway sends the message m_2 unto the ECU that belongs to group. The ECU makes sure that message came from a legitimate gateway using $A_i = E_{K_{Gr_z}}[\overline{ID_S} \| ID_{GW} \| ID_{Gr_z} \| ID_{E_i} \| r \| t_2]$. And if it comes from a legitimate gateway, the ECU believes that the server is the legitimate server. Also, if the server receives message m_3 from the ECU, the server would verify if it came from the legitimate ECU. At the end of verify, if it is correct that the message came from a legitimate ECU, it would achieve mutual authentication between the server and the ECU. Thus, the proposed security protocol could mutual authentication.

Theorem 3. The proposed security protocol could provide a secure session key establishment.

Proof 3. The proposed protocol sets up a session key after the authentication phase. For the session key establishment, the server's secure parameter *r* and the ECU's secure parameter ID_{SK} are required that a malicious user cannot obtain the both for creating the session key. Thus, the proposed security protocol could provide a secure session key setup.

Theorem 4. The proposed security protocol could withstand replay attacks.

Proof 4. The proposed protocol defends a replay attack using timestamp. For that, the proposed protocol uses a timestamp when it creates and receives a message. Suppose a malicious user replays a captured old message to the manufacturer server (i.e., $m_3 = \{L, t_a, ID_{SK}, \upsilon\}$), the vehicle gateway (i.e., $m_1 = \{CID_i, t_1\}$), and the ECU (i.e., $m_2 = \{A_i, t_2\}$). However, the entities find the attack by checking the timestamp of message because new timestamp is applied new message making malicious user (i.e., $t_* - t_1 \le \Delta t$, $t_* - t_2 \le \Delta t$, and $t_* - t_3 \le \Delta t$). So, we could demonstrate that the entity could find the replay attacks. Thus, the proposed security protocol could withstand replay attack.

Theorem 5. The proposed security protocol could withstand an assumed server attack.

Proof 5. Suppose that a malicious user guises the server and forges the message m_1 . Then the malicious user creates a new message that is including itself parameters and sends message to the gateway. However, the gateway does not decrypt the forgery message because the message was not encrypted with the symmetric key between the server and the gateway. The gateway knows that the message came from malicious user. Therefore, the proposed security protocol could withstand the assumed server attack.

Theorem 6. The proposed security protocol could withstand assumed gateway attacks.

Proof 6. A malicious user cannot impersonate the gateway because the user cannot obtain any value (i.e., J, K, K_{Gr_z}) of the gateway. The proposed security protocol thus could withstand assumed gateway attacks.

Theorem 7. The proposed security protocol could withstand gateway secret assume attacks.

Proof 7. The proposed scheme is safe from the gateway secret assume attack. It cannot guess the gateway secret because the gateway does not send the three master keys (i.e., J, K, Q) in the clear. Thus, the proposed security protocol could withstand gateway secret assume attack.

5 Conclusion

In this paper, we have presented a group key based session key establishment protocol for a remote vehicle diagnosis. The proposed scheme aims at providing secure authentication and session key establishment between a vehicle manufacturer server and a group of in-vehicle electronic control unit based on a key graph. The proposed scheme has better key management efficiency than symmetric key systems, while providing lower computation cost than public key systems. As the proposed scheme is generic, it can be extended for securing CCN based vehicular communications [22].

Acknowledgment

Jong-Hyouk Lee is a corresponding author. This work was supported by a 2016 research grant from Sangmyung University.

References

[1] S. Kim, Connected Car Drastic Market Expectations in IoT/M2M Technological Environment, *KISTI Market Report*, Vol. 4, No. 2, pp. 3-6, February, 2014.

[2] M. Asplund, Automatically Proving the Correctness of Vehicle Coordination, *ICT Express*, Vol. 4, No. 1, pp. 51-54, March, 2018.

[3] B.-Y Kang, B. Kyu, W.-C. Seo, E. Yang, D.-W. Seo, Performance Analysis of WAVE Communication under High-speed Driving, *ICT Express*, Vol. 3, No. 4, pp. 171-177, December, 2017.

[4] N. Elgaml, A. Khattab, H.-A. Mourad, Towards Low-delay and High-throughput Cognitive Radio Vehicular Networks, *ICT Express*, Vol. 3, No. 4, pp. 183-187, December, 2017.

[5] G. L. Gragnani, S. Bergamaschi, C. Montecucco, Algorithm for an Indoor Automatic Vehicular System Based on Active RFIDs, *ICT Express*, Vol. 3, No. 4, pp. 188-192, December, 2017.

[6] T. D. T. Nguyen, T.-V. Le, H.-A. Pham, Novel Store-carry-forward Scheme for Message Dissemination in Vehicular Ad-hoc Networks, *ICT Express*, Vol. 3, No. 4, pp. 193-198, December, 2017.

[7] S. Ansari, T. Boutaleb, S. Sinanovic, C. Gamio, I. Krikidis, MHAV: Multitier Heterogeneous Adaptive Vehicular Network with LTE and DSRC, *ICT Express*, Vol. 3, No. 4, pp. 199-203, December, 2017.

[8] T. Cox, P. Thulasiraman, A Zone-based Traffic Assignment Algorithm for Scalable Congestion Reduction, *ICT Express*, Vol. 3, No. 4, pp. 204-208, December, 2017.

[9] J. Greenough, The Connected Car Report: Forecasts, Competing Technologies, and Leading Manufacturers, *BI Intelligence*, January, 2016.

[10] C. Miller, C. Valasek Remote Exploitation of an Unaltered Passenger Vehicle, *DEFCON*, Las Vegas, CA, August, 2015.

[11] C.-M. Yu, C.-Y. Chen, H.-C. Chao, Privacy-Preserving Multi-Keyword Similarity Search over Outsourced Cloud Data, *IEEE Systems Journal*, Vol. 11, No. 2, pp. 385-394, June, 2017.

[12] Y. Nakamura, K. Harada, H. Nishi, A Privacy-Preserving Sharing Method of Electricity Usage Using Self-Organizing Map, *ICT Express*, Vol. 4, No. 1, pp. 24-29, March, 2018.

[13] D. Mashima, A. Serikova, Y. Cheng, B. Chen, Towards Quantitative Evaluation of Privacy Protection Schemes for Electricity Usage Data Sharing, *ICT Express*, Vol. 4, No. 1, pp. 35-41, March, 2018.

[14] A. Ilavendhan, K. Saruladha, Comparative Study of Game Theoretic Approaches to Mitigate Network Layer Attacks in VANETs, *ICT Express*, Vol. 4, No. 1, pp. 46-50, March, 2018.

[15] S. Wi, K. Moon, J.-H. Lee, Symmetric Key Based Session Key Establishment Protocol for Remote ECU Management, *KIISE Winter Conference*, Pyeongchang, Republic of Korea, 2015, pp. 1482-1484.

[16] S. Wi, K. Moon, B. Lee, J.-H. Lee, Group Key Based Session Key Establishment Protocol for a Vehicle Diagnostic, *QSHINE*, Seoul, Republic of Korea, 2016, pp. 62-71.

[17] J. H. Lee, H. Y. Hwan, Vehicle-mounted Ethernet Technology and Standard Trend, *TTA Journal*, Vol. 146, pp. 61-66, March, 2013.

[18] S. Tuohy, M. Glavin, C. Hughes, E. Jones, M. Trivedi, L. Kilmartin, Intra-Vehicle Networks: A Review, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 2, pp. 534-545, April, 2015.

[19] Telecommunications Technology Association, *Secure Requirements and Framework for Multicast Communication*, TTA Standard, TTAK.KO-12.0083/R1, December, 2010.

[20] C. K. Wong, M. Gouda, S. S. Lam, Secure Group Communications Using Key Graphs, *IEEE/ACM Transactions on Networking*, Vol. 8, No. 1, pp. 16-30, February, 2000.

[21] P. Kumar, S.-G. Lee, H.-J. Lee, E-SAP: Efficient-Strong Authentication Protocol for Healthcare Applications Using Wireless Medical Sensor Networks, *Sensors*, Vol. 12, No. 2, pp. 1625-1647, February, 2012.

[22] S. Wang, Z. Yan, G. Geng, Y. Zhang, Geo-based Content Naming and Forwarding Mechanism for Vehicular Networking over CCN, *International Journal of Internet Technology and Secured Transactions*, Vol. 6, No. 4, pp. 291-302, January, 2016.

Biographies

Sarang Wi studied network security from January 2015 to November 2016 at the Protocol Engineering Laboratory (PEL), Sangmyung University. Research interests include authentication and key management.

Jong-Hyouk Lee received his M.S. and Ph.D. degrees in Computer Engineering from Sungkyunkwan University, Suwon, Korea, in 2007 and 2010, respectively. Dr. Lee was a researcher at INRIA, France and was an Assistant Professor at TELECOM Bretagne, France. Since September 2013, he has been with Sangmyung University, Cheonan, Korea. Dr. Lee won the Best Paper Award at the *IEEE WiMob 2012* and received the 2015 Best Land Transportation Paper Award from the *IEEE Vehicular Technology Society*. He was a tutorial speaker at the *IEEE WCNC 2013*, *IEEE VTC 2014 Spring*, and *IEEE ICC 2016*. He was introduced as the Young Researcher of the Month by the *National Research Foundation of Korea (NRF)* in November 2014. He is an associate editor of *Springer Annals of Telecommunications*, *IEEE Consumer Electronics Magazine*, and *IEEE Transactions On Consumer Electronics*. Dr. Lee is an IEEE senior member. Research interests include Blockchain, Mobility Management, Protocol Analysis, Malware Analysis.