Vulnerability verification for illegal attack on secure CAN communication

IOP Conf. Series: Materials Science and Engineering **575** (2019) 012012 doi:10.1088/1757-899X/575/1/012012

Vulnerability verification for illegal attack on secure CAN communication

K Nishida[1*], Y Nozaki[1] and M Yoshikawa[1]

[1] Department of Information Engineering, Meijo University, 1-501 Shiogamaguchi, Tempaku-ku, Nagoya, Aichi 468-8502 Japan.

*150441108@ccalumni.meijo-u.ac.jp

Abstract. Message authentication using MAC has been proposed in order to prevent illegal attacks against CAN which is the standard of in-vehicle communication protocol. It also prevents replay attacks by inserting counter number in MAC. However, it is difficult to synchronize with counter number. Thus, message authentication cannot be performed correctly. Therefore, the counter synchronization mechanism is required. Regarding in-vehicle system, security verification is important as well as safety evaluation. This study proposes a new replay attack focused on the counter synchronization. This study also implements CAN communication with MAC. Experiments using implemented CAN communication show that the proposed replay attack succeeds. Experiment results indicate that the counter synchronization is a key technique to achieve secure in-vehicle system.

1. Introduction

Currently, an automobile is controlled by Electronic Control Unit (ECU). As the advancement of automobile, the number of ECUs is increasing. All ECUs are connected on Controller Area Network (CAN) which is the standard of in-vehicle protocol.

However, the security of CAN is not sufficient, and it is reported that the automobile is illegally controlled [1][2]. In 2015, it was demonstrated that the target vehicle could be controlled illegally and remotely [3]. This dreadful incident caused a massive recall. Therefore, many researches to improve the security of in-vehicle system has been reported. An intrusion detection method for detecting malicious messages is known as example of countermeasure against the attacks [4]. Besides, secure boot has been proposed to detect tampering with ECU [5]. The most typical countermeasure is to perform message authentication by Message Authentication Code (MAC). In secure CAN communication using MAC, MAC is generated including the nonce by the counter because simply generated MAC is subject to the replay attack. On the other hand, if the counter is out of synchronization, there is a possibility that authentication cannot be performed correctly. Therefore, in the authentication method proposed in [6], the global nonce generator is introduced. The nonce generator periodically broadcasts a random global nonce, and the counter is resynchronized.

Since attacks on in-vehicle system are threat to life, to enhance security of CAN is very important. Therefore, security evaluation of secure CAN communication is necessary.

This study proposes a new attack method on CAN communication with MAC. Actually, in CAN communication using counter value for MAC generation, the resynchronization mechanism of the counter is required. The proposed method performs replay attack focused on a synchronization message of the counter. Experiments using mock-up system, which is implemented CAN

communication with MAC generated from control data and counter value, show that the proposed method succeeds. Experiment results also indicate that synchronization of the counter needs to be performed securely.

2. Preliminaries

2.1. Controller Area Network

CAN [7] is a communication protocol used in a vehicle network. CAN transmit data using a potential difference between two signal lines. CAN is resistant to noise because the noise applied to the two signal lines is the same and the potential difference does not change. CAN protocol adopts multi-master system, and when the CAN bus is free, communication can be started from any node. If the multiple nodes start communication at the same time, high priority message is transmitted. In the arbitration, the smaller ID has priority over the larger one. CAN communicates by using four kinds of frames called data frame, remote frame, error frame, and overload frame. Figure 1 shows the format of the data frame. The data frame is a format used in data transmission, and data is stored in the data field of the data frame. The maximum length of the data field is 64 bits.

2.2. Message Authentication Code

MAC is the information for message authentication. Message authentication is to check whether the message is tampered. MAC is generated from the shared key and the message. Figure 2 shows the overview of the message authentication. The sender and the receiver share key before starting communication. First, the sender generates MAC from the shared key and the data. The sender sends the data and generated MAC to the receiver. Then, the receiver generates MAC in the same way as the sender. If the generated MAC is equal to received MAC, the sent data is not tampered.

2.3. Cipher-based Message Authentication Code (CMAC)

CMAC [8] is MAC based on block cipher. Figure 3 shows the two cases of MAC generation process in the CMAC. Figure 3 (i) shows the case that the length of M_n fulfills the block size b, and figure 3 (ii) shows the other case. The subkeys K_1 and K_2 are generated from the shared key. The message M is divided into M_1 to M_n by the block size b. As shown in figure 3 (ii), padding is performed on M_n when the length of M_n does not fulfill the block size. MAC is generated by performing encryption ENC_k and exclusive OR operation. The generated MAC length is equal to the block size b.

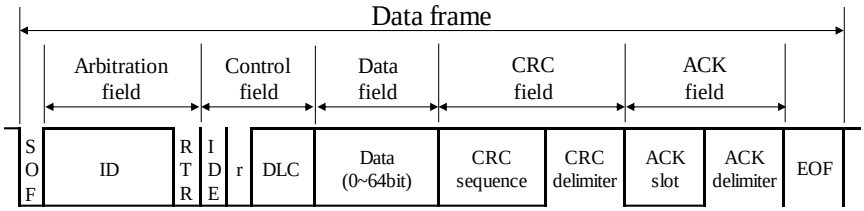

Figure 1. Data frame format.

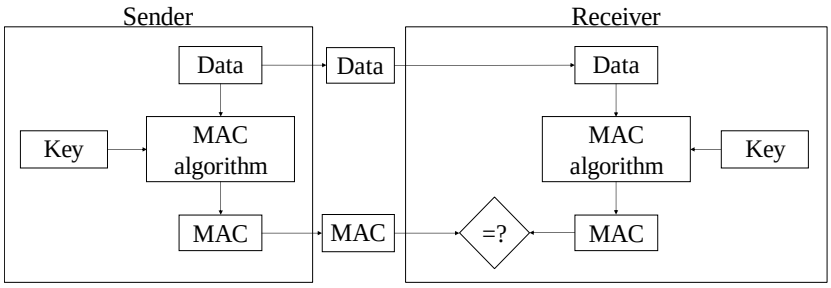

Figure 2. Overview of the message authentication.

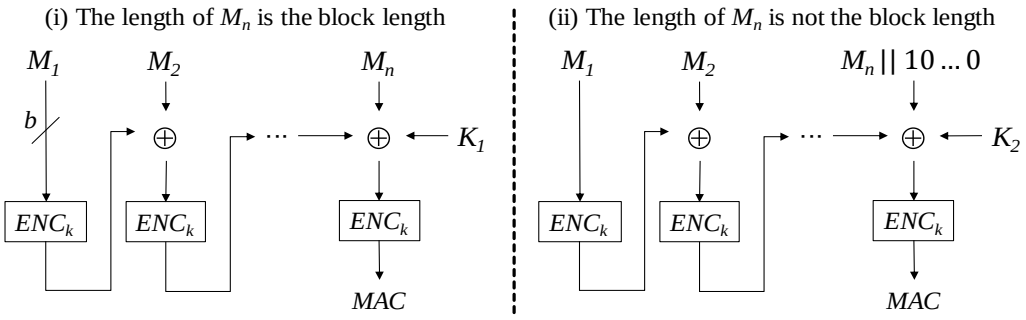

Figure 3. MAC generation processes in CMAC.

3. Proposed method

3.1. Countermeasure of in-vehicle system

This study applies AES-CMAC [9] to CAN communication. AES-CMAC is an authentication algorithm based on CMAC with Advanced Encryption Standard (AES). Figure 4 shows the overview of the implemented CAN communication with MAC. The sending side and the receiving side hold the secret key and the counter. The secret key is shared before starting communication.

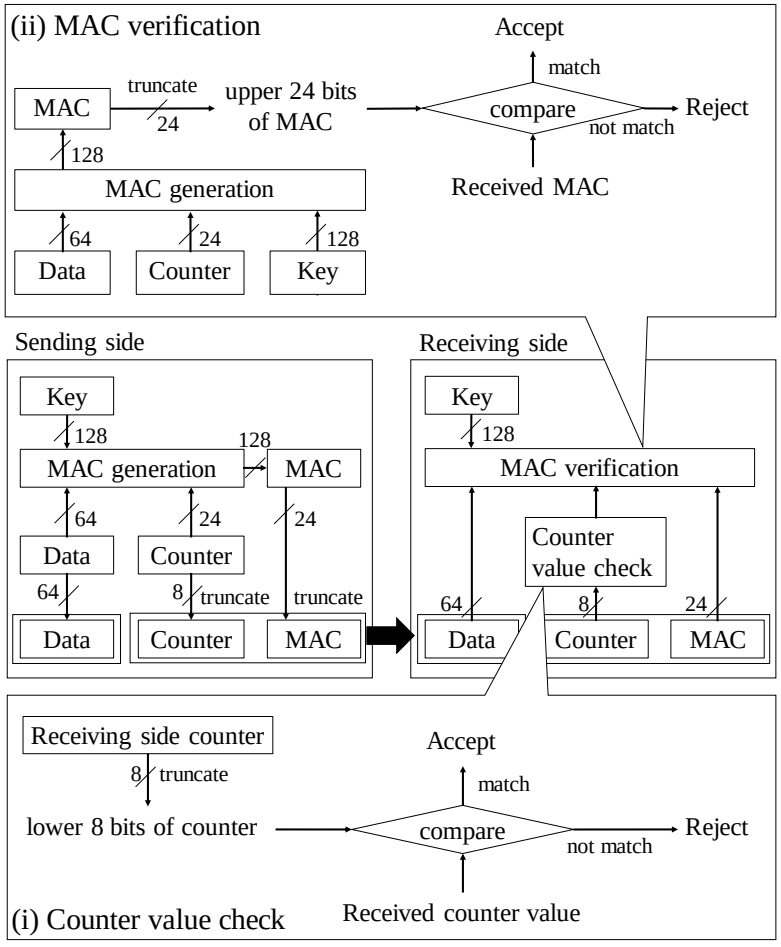

Figure 4. Overview of the implemented CAN communication.

First, the sending side or the receiving side increments each counter at the time of sending or receiving control data. Next, as shown in figure 4, on the sending side, MAC is generated from a 128-bit key, a 64-bit control data, and a 24-bit counter. At this time, the generated MAC length is longer than the data field length of data frame. Therefore, part of the generated MAC is truncated, and the upper 24 bits are treated as MAC. Figure 5 shows messages format. Then, the sending side sends the lower 8 bits of the counter value and the MAC after sending the control data. Also, the receiving side receives the 8-bit truncated counter value and the 24-bit MAC after receiving the control data. Then, counter value check part compares the received counter value and the lower 8 bits of the counter value on the receiving side. When the counter values do not match, the control data is discarded, as shown in figure 4 (i). Subsequently, the receiving side generates MAC in the same way as the sending side. Finally, MAC verification part compares the received MAC and the upper 24 bits of the generated MAC, as shown in figure 4 (ii). When the MACs do not match, the control data is discarded.
The synchronization of the counter is performed by CAN message. When the synchronization message is received, ECU sets the held counter to zero.

3.2. Attack procedure
This study assumes that an attacker can capture messages on the CAN bus and freely send messages. Figure 6 shows the overview of the proposed method. The proposed method performs replay attack focused on synchronization message of the counter. First, an attacker observes CAN bus, and captures synchronization message of the counter and subsequent messages (see figure 6 (i)). Then, the attacker sends the captured messages to CAN bus (see figure 6 (ii)). Actually, the implemented system does not perform to authenticate synchronization message. As a result, the messages retransmitted from the attacker can be authenticated because the counter is reset by the synchronization message included in captured messages.

4. Evaluation experiments

4.1. Experiments environment
Figure 7 shows experiments environment. This study used two M32C boards as ECU for sending and receiving side. The CAN communication using AES-CMAC (see Sect. 3.1.) is implemented on the M32C boards. The control data is transmitted from the controller to the sending side. In addition, the control data is displayed on the 7 segment LED on the M32C boards. In 7 segment LED, when authentication succeeds, the displayed control data is updated. This confirms that the messages retransmitted from the attacker is authenticated. For the observation of CAN messages, the MicroPecker CAN Analyzer was used.
Table 1 shows data contents of CAN messages in the implemented CAN communication. The synchronization message with ID 010 is transmitted by pushing button on the M32C board to perform easily experiments. The messages with ID 101 and 103 are control data and the messages with ID 102 and 104 are data for authentication. As the counter needs to be synchronized for authentication, the high priority ID 010 was assigned to the synchronization message.

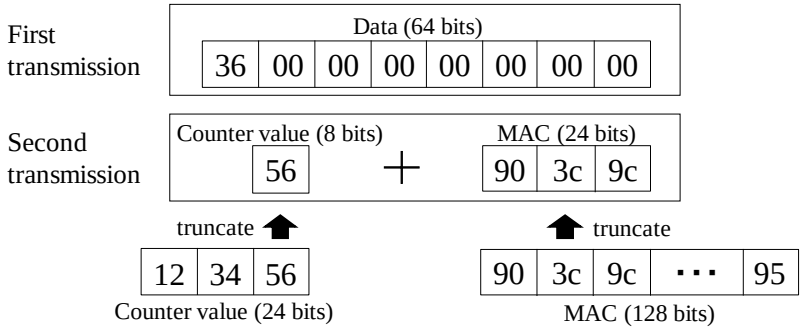

Figure 5. Message format in the implemented CAN communication.

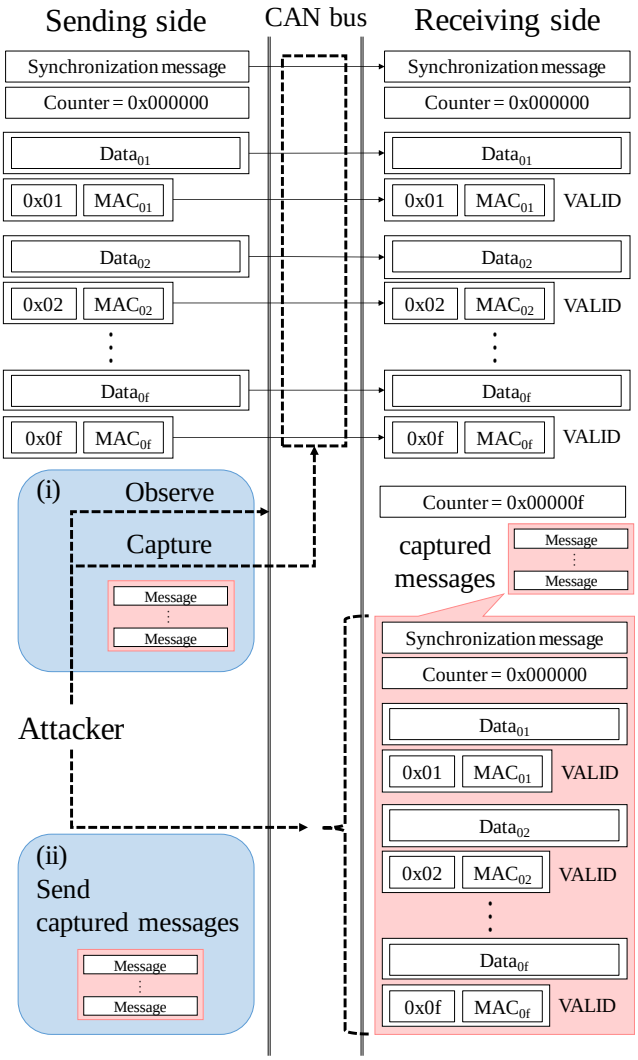

Figure 6. Proposed method.

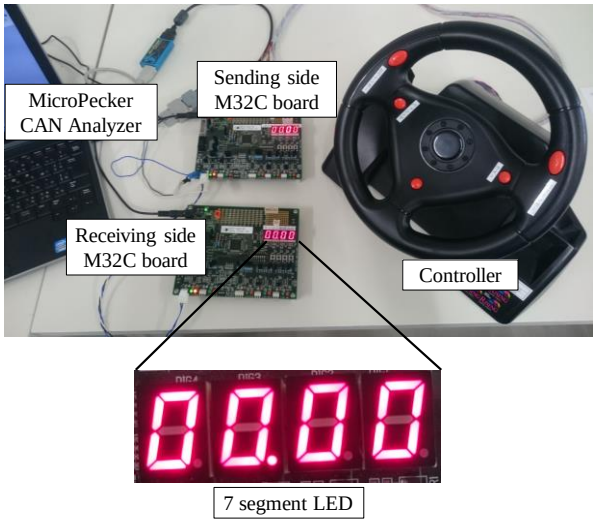

Figure 7. Experiments environment.

Table 1. Data contents of CAN messages.

ID	Length	Content
010	3 bytes	Synchronization message
101	8 bytes	Control data
102	4 bytes	Counter value and MAC with ID 101
103	8 bytes	Control data
104	4 bytes	Counter value and MAC with ID 103

4.2. Experiment results

First, this study confirmed that the implemented CAN communication communicated correctly. Figure 8 shows observation result of the implemented CAN communication. As shown in figure 8, counter value and MAC were transmitted after the control data. 7 segment LED on the receiving side displayed transmitted control data. As a result, communication was performed normally in the implemented CAN communication.

Next, the normal replay attack was conducted. In the normal replay attack, an attacker retransmitted only one set of control data, counter value and MAC. Figure 9 shows observation result of the normal replay attack. In this figure, "5C 00 00 00 00 00 00 00", "57", and "72 41 F9" represent the retransmitted control data, counter value, and MAC, respectively. On the receiving side, the retransmitted control data was not displayed on 7 segment LED because different counter values were used each time for MAC generation. Thus, the implemented CAN communication can prevent the normal replay attack.

Finally, the proposed method was conducted. Figure 10 shows the results by the proposed method. In figure 10, the message with ID 010 is the synchronization message of the counter. At this time, the counter of the receiving side was reset to zero by this message. Therefore, the retransmitted messages after synchronization message were authenticated because the counter value of the receiving side synchronized that of the retransmitted messages. Also, on the receiving side, retransmitted control data was displayed on 7 segment LED. Thus, the proposed method can succeed in illegal attack. Therefore, countermeasure against replay of synchronization message is necessary.

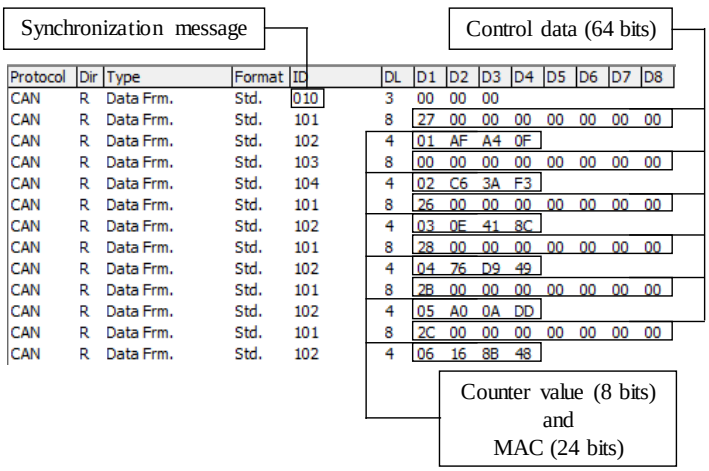

Figure 8. Observation result of the implemented CAN communication.

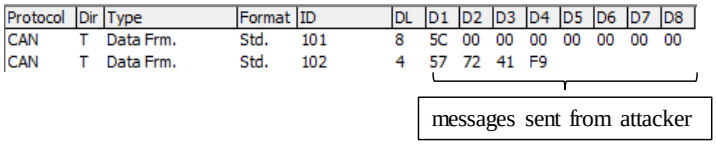

Figure 9. Observation result of the normal replay attack.

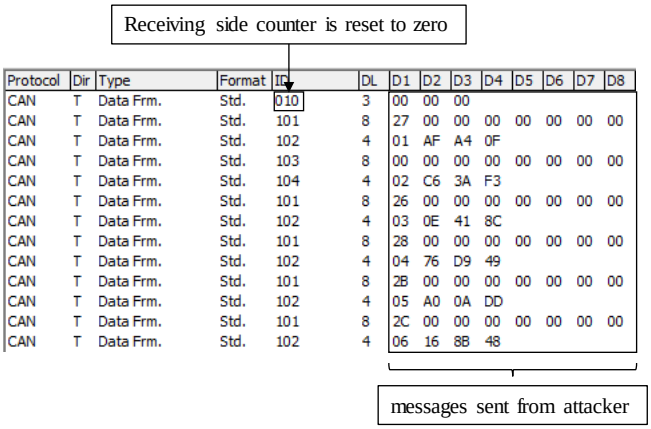

Figure 10. Observation result of the proposed method.

5. Conclusions

This study proposed a new replay attack focused on the counter resynchronization of CAN communication with MAC generated from control data and counter value. The proposed method can attack by control data in the retransmitted message on the condition that attacker can capture and send messages. This study also implemented CAN communication using AES-CMAC. Experiments using the implemented CAN communication showed that the proposed method succeeded and countermeasure against replay of synchronization message was necessary.
Future works include study of secure synchronization of counter.

6. References

[1] Koscher K, Czeskis A, Roesner F, Patel S, Kohno T, Checkoway S, McCoy D, Kantor B, Anderson D, Shacham H and Savage S 2010 Experimental Security Analysis of a Modern Automobile *Proc. 2010 IEEE Symp. on Security and Privacy* pp 447–462
[2] Miller C and Valasek C 2014 Adventures in Automotive Networks and Control Units https://ioactive.com/pdfs/IOActive_Adventures_in_Automotive_Networks_and_Control_Units.pdf
[3] Miller C and Valasek C 2015 Remote Exploitation of an Unaltered Passenger Vehicle http://illmatics.com/Remote%20Car%20Hacking.pdf
[4] Kuwahara T, Baba Y, Kashima H, Kishikawa T, Tsurumi J, Haga T, Ujiie Y, Sasaki T and Matsushima H 2018 Supervised and Unsupervised Intrusion Detection Based on CAN Message Frequencies for In-vehicle Network *J. of Information Processing* **26** pp 306–313
[5] Takemori K, Mizoguchi S, Kawabata H and Kubota A 2016 In-Vehicle Network Security Using Secure Element *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences* **E99-A** 1 pp 208–216
[6] Nürnburger S and Rossow C 2016 – vatiCAN – Vetted, Authenticated CAN Bus *Proc. Cryptographic Hardware and Embedded Systems – CHES 2016* pp 106–124
[7] Robert Bosch GmbH 1991 CAN Specification
[8] Dworkin M 2005 Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38b.pdf
[9] Song JH, Poovendran R, Lee J and Iwata T 2006 The AES-CMAC Algorithm https://tools.ietf.org/html/rfc4493