EFFICIENT MESSAGE AUTHENTICATION CODE (MAC) BASED CYBER SECURITY FOR VEHICLE NETWORK

[1]Pavithra M J, [2]Nirmala, [3]Shankara C

[1]Lecturer, [2] Lecturer, [3]Lecturer
[1,2,3]Department of Electronics and Communication Engineering
[1,2]Government Polytechnic K.R. Pet, Mandya, India,
[3]Government Polytechnic Nagamangala, Mandya, India

Abstract: We are living in a world that is becoming increasingly networked on a daily basis, and we are witnessing a global transition in which all of the things that are in our immediate environment are becoming "smart" and connected to the Internet. In addition, the automotive industry is a component of this transformation. An in-vehicle network is created by the presence of more than 150 electronic control units (ECUs) in modern automobiles. This network is responsible for controlling the functions of the vehicle, similar to the fly-by-wire concept used in aviation. Each of these automobiles is equipped with many Internet connection points, allowing them to provide their passengers with a wide range of online services. Even while any item that connects to the Internet is vulnerable to a variety of online dangers, the linked automobile is also vulnerable to these dangers. As a result, it is becoming increasingly important to address the concerns regarding cyber security that are prevalent in the automobile industry. Electronic control units (ECUs) of vehicles are connected via several communication busses on the inside. Reading and sending data to other ECUs is possible for any ECU that is connected to the bus. As a consequence of this, if an adversary is successful in compromising one of the ECUs, then the adversary will be able to access and exploit the data of data stored on other significant ECUs. The primary reason for this is that there is no confidentiality about the matter. Furthermore, the communications are more susceptible to being compromised because there is a lack of either data integrity or authenticity. It has been demonstrated in the past that an enemy can take control of the vehicle by taking advantage of the inadequacies of CIA, which stands for confidentiality, integrity, and authenticity. If the integrity of an essential ECU is breached, an attacker has the ability to alter the data that is stored on the device. In order to address these issues, the purpose of this study is to propose the design of a message authentication framework that would enable the internal communication to be secured in an effective manner. Due to the fact that the calculation of the message authentication code is dependent on the Authentication Data, Secured Key, CMAC algorithm, and extra counters, the likelihood of a hacker mimicking a message to the vehicle network is significantly reduced. In addition to this, a test technique for the validation of the message authentication code is presented.

Index Terms - **Control Area network (CAN) In-vehicle network, data authentication, message authentication code (MAC), CMAC Algorithm, ECU, CANoe, DaVinci Developer.**

I. INTRODUCTION

As a result of the constant stream of new features that are designed to make driving more enjoyable, the number of electronics that are used in the automotive industry has significantly expanded. Automobiles have evolved into intricate mechanical systems that can have as many as eighty Electronic Control Units (ECUs) for the most expensive sector of the market. Consequently, in-car networks are becoming increasingly complicated in order to allow communication among the numerous devices that are now in use.It appears that communication between automobiles and infrastructure, as well as communication between automobiles themselves, will be a plausible aspect of the Internet of Things (IoT) paradigm in the next years. In light of the rising number of access points that are available to a possible attacker, the automotive sector ought to therefore provide secure communication in order to safeguard the data that is transporting via networks. There is no doubt that the information contained inside automotive networks is frequently associated with or interacts with human safety. A lack of security can have a negative influence on health or even result in the loss of lives. Additionally, the growing connectedness of automobiles with the surrounding environment may also have an effect on the transfer of personal data, the exposure of which may have repercussions for either the individual's private life or their financial situation.

The automobile industry has, for many years, solely regarded security in a physical sense, meaning that it was only concerned with ensuring that a car could not be stolen or broken into. On the other hand, this has been dramatically shifting over the course of the past ten years: computer security, which is often referred to as cyber-security, is now included in the scope of automobile security [1,2]. Additionally, it is important to understand that security is not the same thing as safety. Despite the fact that the overarching objective of security and safety is the same, namely the protection of the system and the humans operating within the system's environment, the fault model that underpins these two concepts is distinct. Security is typically concerned with the protection against intentional malicious manipulation, whereas safety is concerned with the protection against random faults [3,4].

The automobile industry has seen a significant increase in the relevance of cyber-security over the course of the past decade, and this can be ascribed to a number of different sources. Many automotive systems that were once mechanical or analog are now digital, including safety-critical operations such as steering and braking [5]. This is one of the elements that contributes to the growth of technology in the automobile industry. The implementation of digitalization results in reduced production costs, simplified maintenance, and the ability to perform complex signal processing on very straightforward technology. On the other hand, given that the majority of functions are programmable and controlled by software, it also makes it possible for malevolent manipulation to occur [6].

There is a bigger trend in society and across all industries to interconnect all different kinds of devices in order to permit new kinds of services, which is another aspect that is contributing to the increased interest in security. The automotive industry is no exception: vehicles connect to “cloud” services, for instance for remote diagnostics or remote software upgrades, and customer expectations are that gadgets such as smartphones integrate seamlessly with the vehicle [7,8,9]. The result is that attackers have access to a substantially greater attack surface, which in turn increases the requirement for security. The development of what are known as intelligent transportation systems (ITS) will be the last factor that we will take into consideration. Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication are two examples of the completely new communication channels that are being introduced by intelligent transportation systems (ITS) that are now being developed to improve road safety and traffic flow [10]. Because of this, there is a potential for danger in the event that the communication infrastructure and the systems that are participating are not adequately safeguarded.

When it comes to the in-vehicle network, the controller area network (CAN) bus, which is responsible for the control and maneuverability of the vehicle, is the most obvious target for an attacker. On the CAN bus, it is feasible to carry out a number of different attack activities, including read, inject, and modify, as demonstrated in [11]. In light of the fact that these messages have the potential to influence the control of the vehicle, we contend that the preservation of data integrity and data authentication is more important than the protection of data secrecy. Therefore, it is vital to have appropriate authentication measures, such as message authentication codes (MAC), in order to safeguard against the possibility of an adversary injecting and changing messages within the CAN bus itself.

In the sense that it is a real-time system that is comprised of embedded ECUs that are limited in terms of resources, the in-vehicle network is a nontraditional environment. As a result, availability is an extremely critical security attribute. Consequently, the conventional approaches to data authentication are not appropriate in this context. Across the addition of a message authentication code (MAC) to the data that is going to be transmitted across the CAN bus, this research presents an effective method of ensuring the safety of the vehicle.

II. Related work

The in-vehicle network has been the subject of many studies in the past, the majority of which have concentrated on safety features. It is only relatively recently that studies have begun to investigate the security of this network. The majority of research has concentrated on vulnerabilities and assaults on the CAN protocol. This is due to the fact that the CAN bus is a potential target in the in-vehicle network.

In their work [12, 13], Nilsson and colleagues demonstrate a variety of simulated attacks on the CAN and FlexRay buses that are part of the in-car network. Additionally, they offer the concept of a vehicle virus. We have taken note of the fact that there is no protection for authentication or integrity.

An algorithm for the identification of malicious CAN messages has been outlined, and it has been implemented in the simulation environment of CANoe. An upgrade of CAN to control system security was presented by T. Hoppe and colleagues [18]. The outcome demonstrates that this algorithm in question possesses a robust detection function and is worthy of being put into practice. In industries such as automobiles, industrial automation, industrial equipment, aircraft, medical equipment, and many others, CAN control systems have been utilized for a considerable amount of time. When it comes to CAN control systems, the most important components have always been the protocols, the hardware, and the software. An algorithm for the identification of malicious CAN messages should be designed, and features of the CAN protocol should be analyzed. Error bits cannot be destroyed, a network problem is raised, and the length of data is restricted. These are some of the limitations.

Wolf et al. [14] discuss a number of challenges that are associated with the CAN and FlexRay bus protocols. These problems include issues connected to authenticity and secrecy. These two groups of researchers, Hoppe et al. [15] and Lang et al. [16], explore the ramifications of simulated sniffer and replay attacks on the CAN bus. They also detail the assaults themselves.

They proposed that data authentication could be helpful in detecting and recovering from injection and alternative assaults in the in-vehicle network. This was done by B. Groza and S. Murvay [17]. An example of a bus that is frequently utilized by controllers found inside of automobiles and in a variety of industrial applications is the Controller Area Network. In this article, we develop a broadcast authentication protocol that is based on key based and time synchronization. This is a way that is often used in wireless sensor networks. This method enables us to benefit from the utilization of symmetric primitives without the requirement of secret keys during the broadcasting process.

III. BACKGROUND

In this part, the CBC-MAC mode of operation is provided, as well as a description of the in-vehicle network and the CAN protocol.

A. In-Vehicle Network

A number of electronic control units (ECUs) are included in the in-vehicle network. These ECUs are connected to a variety of buses through the use of different proto-cols. This study focuses on the Controller Area Network (CAN) protocol because it is used for communication on the powertrain network, which is accountable for the control and maneuverability of the vehicle. Due to the fact that engine control units (ECUs) are highly constrained in terms of their computational capability and data capacity, automobile manufacturers are working hard to keep the costs of the vehicle production process as low as possible. 1 MB of SRAM, 2 MB of flash memory, and a 12 MHz 16-bit processor are the standard components that make up an ECU setup [19, 20].

B.CAN Protocol

The CAN protocol often operates at either 125 kbps or 250 kbps, depending on the circumstances. In addition, the protocol functions in real time, and the processing of messages begins as soon as they are received [21]. Because of this, there is no room for delays or queueing of messages. Due to the fact that these real-time constraints need the prompt processing of messages, the design of security solutions is additionally complicated. Given that it is not possible to utilize heavy cryptographic methods, it is necessary to find solutions that are both unobtrusive and lightweight. On top of that, the traffic patterns on the in-vehicle CAN bus are fairly deterministic; messages are transmitted at regular intervals, such as once every 20 milliseconds.

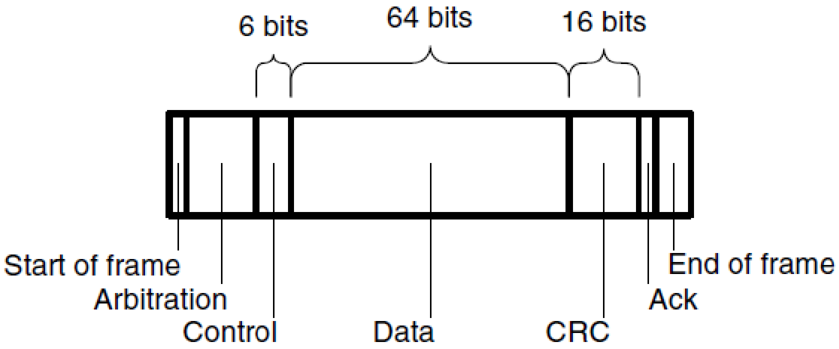


Figure. 1. A CAN data frame equipped with a control field of six bits, a data field of sixty-four bits, and a CRC field of sixteen bits.

The four unique types of frames are data frames, distant frames, error frames, and overload frames. Data frames consist of data frames. In light of the fact that it is responsible for transmitting information from the sender to the receiver, we give the data frame this priority. From what can be seen in Figure 1 [21], the data frame is partitioned into a variety of fields that are distinct from one another. Some of the fields that are utilized include a control field that is composed of six bits, a data field that is composed of sixty-four bits, and a CRC (cyclic redundancy code) field that is composed of sixteen bits. There are two bits in the control field that are not being utilized, and there are four bits that show the amount of bytes that are contained in the data field. Both those bits are located in the control field. It is possible for the data field to store up to eight bytes of information, which is where the information that is going to be sent is kept. The CRC field does not provide any authentication or integrity of the data; rather, it is a simple checksum that is utilized for the purpose of identifying errors that occurred during transmission.

C.CBC-MAC

Block ciphers have a method of encryption known as CBC-MAC, which stands for cipher-block chaining message authentication code. In this mode, the most recent encrypted output is utilized as the message authentication code. This means that the final ciphertext is dependent on all of the plaintexts that came before it since in CBC, all plaintexts are XORed with the block of ciphertext that came before it. A message's message authentication code (MAC) can be computed with minimal effort by employing an effective block cipher. As long as the encryption technique that is used to encrypt the message is secure, Bellare et al. have demonstrated that this mode of encryption for fixed-length communications is secure [22, 23]. The fact that CBC-MAC is an effective approach for calculating MACs is the primary reason for our selection of this particular method.

IV. PROBLEM STATEMENT

Develop an efficient Message Authentication Code (MAC) system tailored for vehicle networks to ensure secure communication. The solution should minimize computational overhead while providing robust protection against unauthorized access and data tampering. It must accommodate the dynamic nature of vehicular environments and support rapid message verification. The MAC system should integrate seamlessly with existing vehicle network architectures and protocols to enhance overall cybersecurity measures.

V. PROPOSED METHODOLOGY

a. Overview

This paper proposes the Efficient MAC based security system for in-vehicle network which includes Gateway, Sending and Receiving ECU's as shown in figure 2.

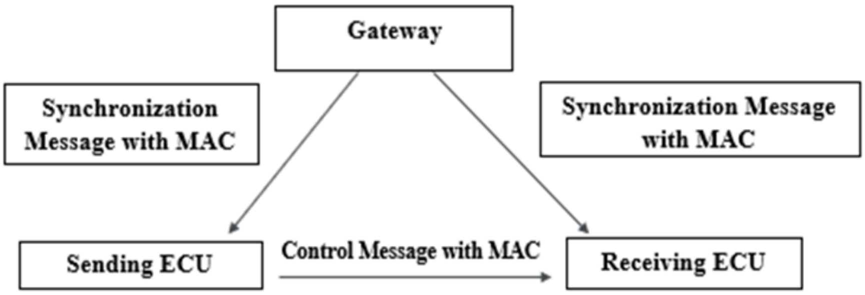

Gateway
Synchronization Message with MAC
Synchronization Message with MAC
Sending ECU
Control Message with MAC
Receiving ECU

Figure 2: Structure of Proposed MAC security system.

Gateway is mainly used to provide synchronization between ECUs in the Vehicle’s Networks. The communication between Gateway and the ECUs is established by using synchronization message with MAC. Each Synchronization Message had a MAC of 128 bit, which is generated by procedure indicated in the figure 3. Message which is transmitted from sending ECU to receiving ECU is called Control Message. Each Control Message had a MAC of 128 bit which is generated by procedure as shown in the figure 3. And Specification used for MAC generation is indicated in the Table 1.

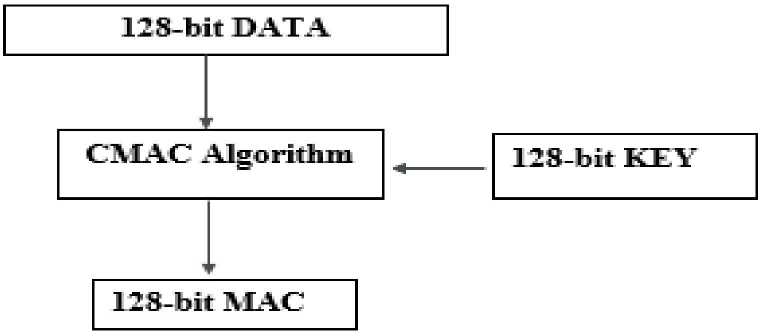


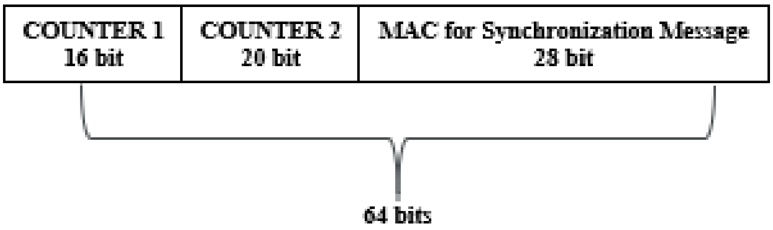
Figure 3 : Overview of MAC generation

Table 1: Specification for MAC generation

b. Construction of data

1. Synchronization Message.

Synchronization message is requirement which needs to be processed at both sending ECU and receiving ECU.which includes two counters along with 128 bit MAC in order to increase Complexcity of the message authentication process while generating the sychronization message at the gateway.



Algorithm	AES
Key Length	128 bits
Length of Block	128 bits
MAC Generation Algorithm	CMAC
MAC Length	128 bits

Figure 4: Message format of synchronization message

Counter 1 and 2 is 16 bit and 20 bit respectively is designed in order to increase the complexity of the overall system. The generation of 128-bit MAC is shown in the above figure 3 and only upper 28 bit of MAC is appended in synchronization message while transmitting from gateway.128 bits of data is input to CMAC algorithm to generate the MAC while validating the synchronization message at the receiving side. Below figure 5 shows the construction procedure of 128 bit data.

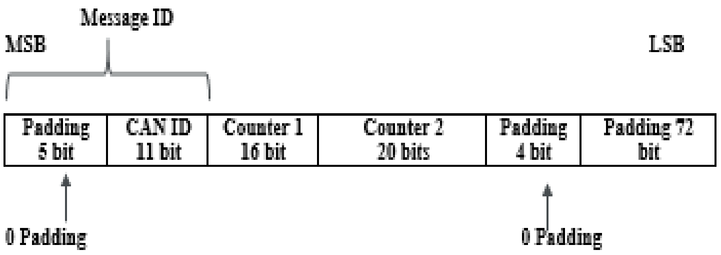


Figure 5: Construction of 128-bit data for Generating MAC while processing synchronization message.

2. Control Message

Control message is costructed by appending the upper 28 bits of 128 bits MAC to the Counter information along with the data to be protected. Below figure 6 shows the structure of control message with MAC.

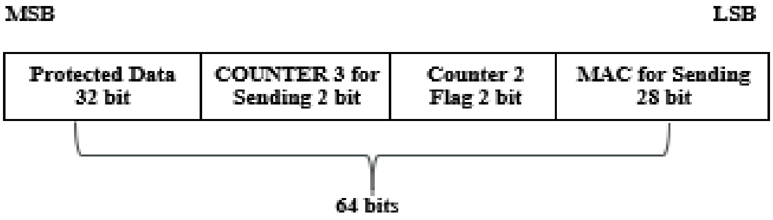


Figure 6. Message format of control message

128 bits of data is input to CMAC algorithm to generate the MAC while validating the control message at the receiving side. Below figure 7 shows the construction procedure of 128 bit data.

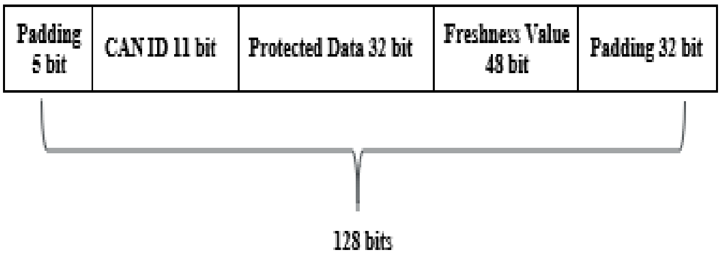


Figure 7: Construction of 128-bit data for Generating MAC while processing control message

c. Process Flow

Sending ECU must execute process flow presented in below figure 8 before sending message to CAN network.

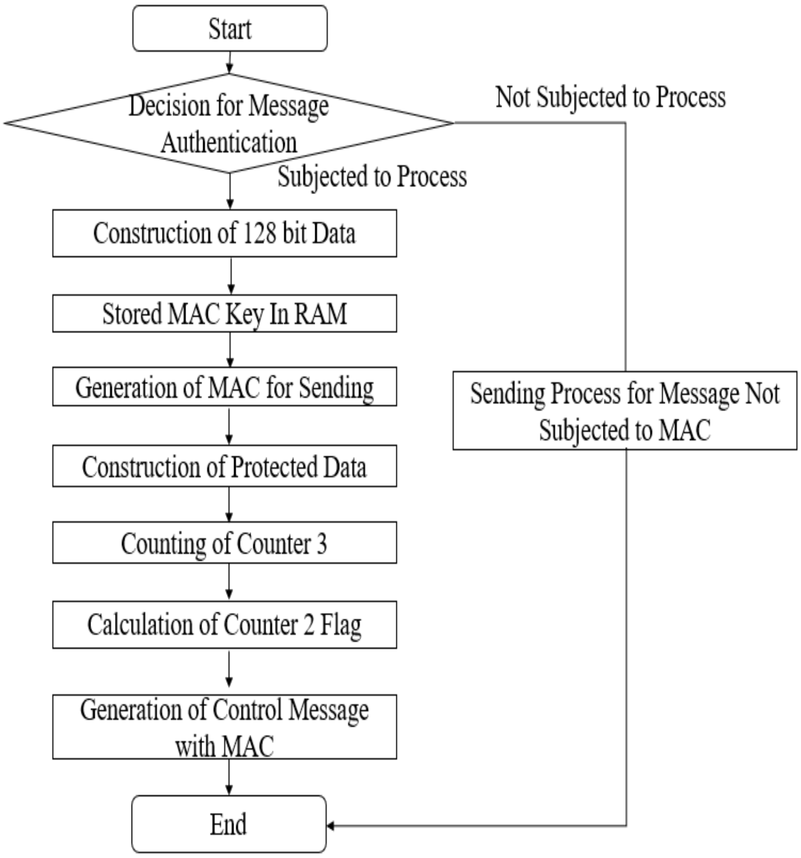

Start
Decision for Message Authentication
Not Subjected to Process
Subjected to Process
Construction of 128 bit Data
Stored MAC Key In RAM
Generation of MAC for Sending
Construction of Protected Data
Counting of Counter 3
Calculation of Counter 2 Flag
Generation of Control Message with MAC
Sending Process for Message Not Subjected to MAC
End

Figure 8: Processing flow for sending control message with MAC.

If receiving ECU has received control message with MAC, it should execute processing flow presented in figure 9.

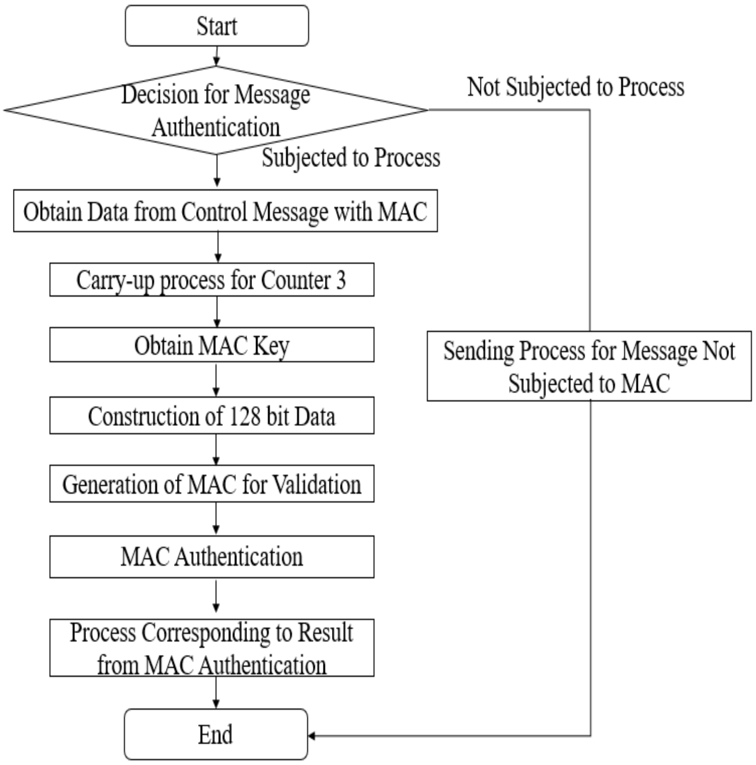

Start
Decision for Message Authentication
Not Subjected to Process
Subjected to Process
Obtain Data from Control Message with MAC
Carry-up process for Counter 3
Obtain MAC Key
Construction of 128 bit Data
Generation of MAC for Validation
MAC Authentication
Process Corresponding to Result from MAC Authentication
Sending Process for Message Not Subjected to MAC
End

Figure 9: Processing flow for receiving control message with MAC

d. MAC Validation

MAC validation is done by comparing MAC included in received synchronization message and 28 bit taken from the most significant bit of MAC generated by the received unit. The same as shown in figure 10.

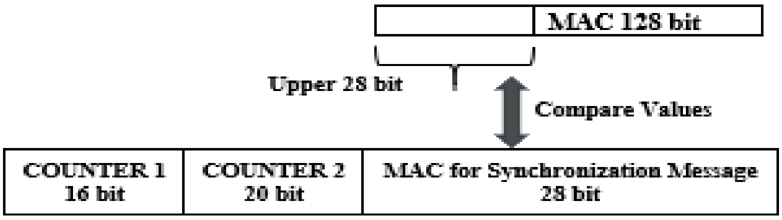


Figure 10: MAC validation for synchronization message

Figure 11 shows the MAC validation flow for control message**.**

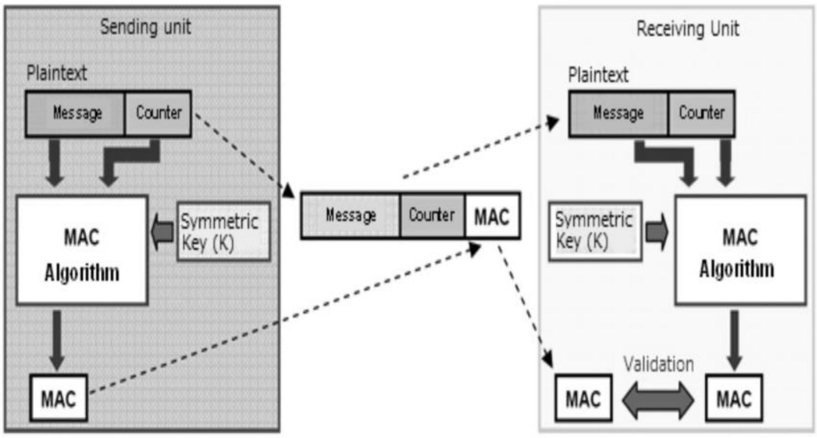


Figure 11: MAC validation for Control message

VI. RESULTS AND DISCUSSIONS

In-vechile CAN network is developed by using CANoe software simulation tool and the proposed security system is created by considering Gateway as one ECU.Messgae Authintication frame work is devloped by using Devinci configurator tool and flashed on to the real ECU as shown the figure 12. The same secret key is shared to both sending and receving ECU's and stored in memeory of ECU.

Message with ID 3D5 is a synchronization message which is sending from the Gateway and this synchronization message is must to transmits and receptions of control messages. When synchronization message 3D5 transmitted from gateway all the control messages transmitted and received at the real ECU. If its 3D5 message stops, all other control messages like message ID 11E, 11D, 11F and 22 B will also stop. same can be observed in CANalyzer and same as shown in Figure 13. MAC validation is done with help of Message with ID 760, this message is indicated the MAC verification results in the form of Flag namely MAC_OK and MAC_NotOK. When MAC Validation done successfully MAC_OK Flag will be set to High and for MAC validation failure MAC_NG flag will be High

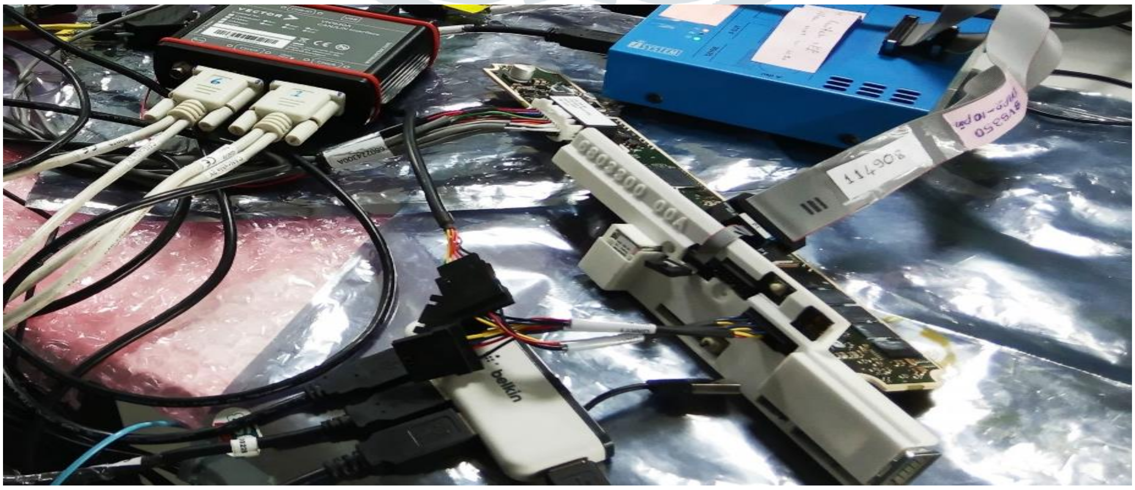
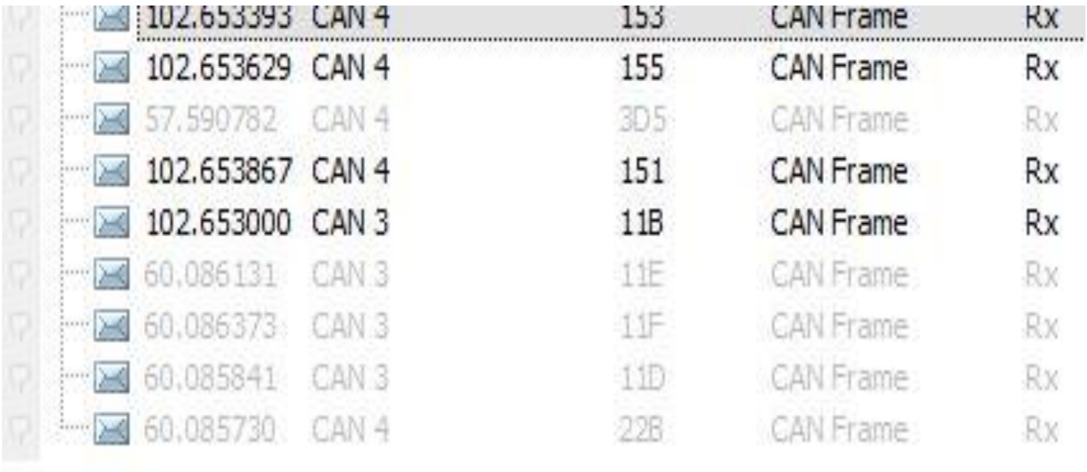


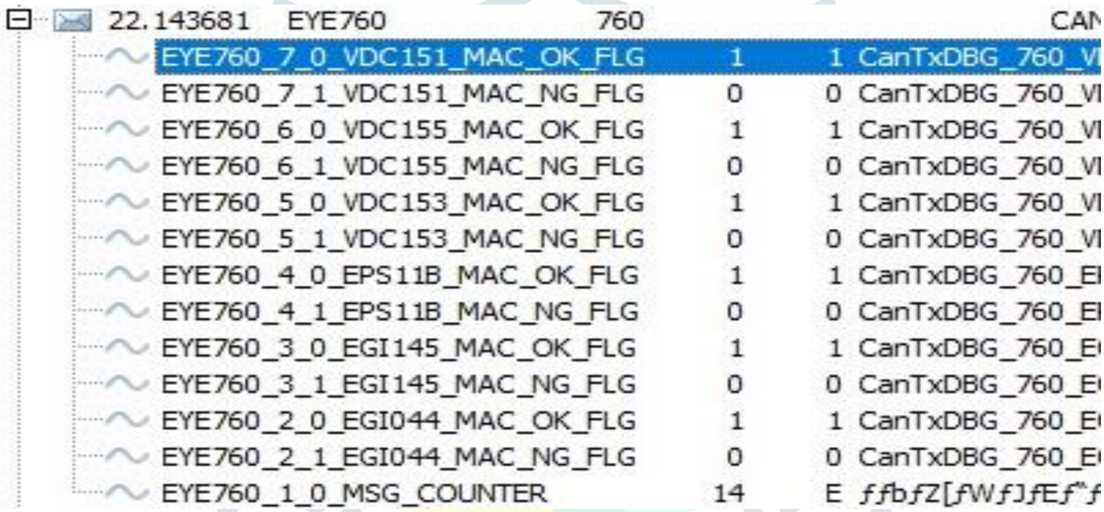
Figure 12: Physical ECU Simulation Setup.



102.653393	CAN 4	153	CAN Frame	Rx
102.653629	CAN 4	155	CAN Frame	Rx
57.590782	CAN 4	3D5	CAN Frame	Rx
102.653867	CAN 4	151	CAN Frame	Rx
102.653000	CAN 3	11B	CAN Frame	Rx
60.086131	CAN 3	11E	CAN Frame	Rx
60.086373	CAN 3	11F	CAN Frame	Rx
60.085841	CAN 3	11D	CAN Frame	Rx
60.085730	CAN 4	22B	CAN Frame	Rx

Figure 13: When Synchronization messages Stops.

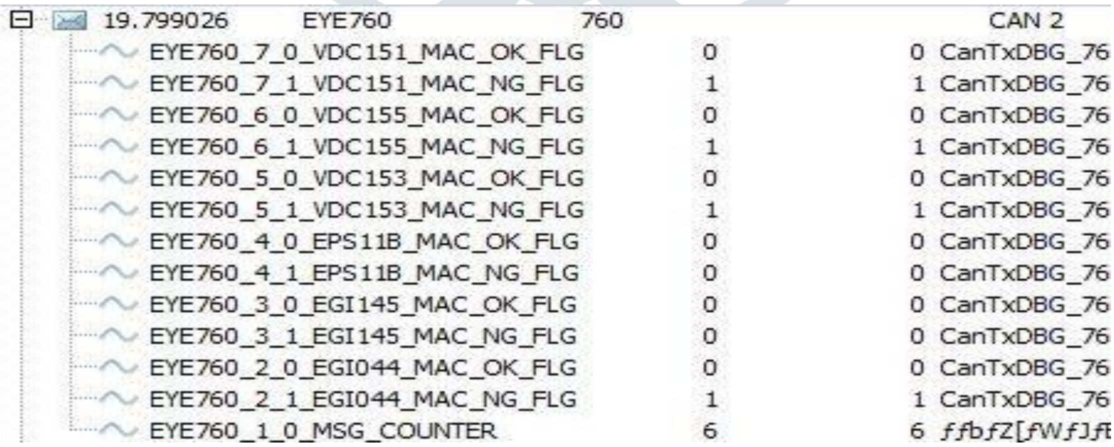
When Receive control message is not received and there will be no MAC validation is done and both the Flags MAC_OK and MAC_NG has been set to High for various control messages, this can be observed in Figure 14.



22.143681 EYE760	760		CAN
EYE760_7_0_VDC151_MAC_OK_FLG	1	1	CanTxDBG_760_VI
EYE760_7_1_VDC151_MAC_NG_FLG	0	0	CanTxDBG_760_VI
EYE760_6_0_VDC155_MAC_OK_FLG	1	1	CanTxDBG_760_VI
EYE760_6_1_VDC155_MAC_NG_FLG	0	0	CanTxDBG_760_VI
EYE760_5_0_VDC153_MAC_OK_FLG	1	1	CanTxDBG_760_VI
EYE760_5_1_VDC153_MAC_NG_FLG	0	0	CanTxDBG_760_VI
EYE760_4_0_EPS11B_MAC_OK_FLG	1	1	CanTxDBG_760_EI
EYE760_4_1_EPS11B_MAC_NG_FLG	0	0	CanTxDBG_760_EI
EYE760_3_0_EGI145_MAC_OK_FLG	1	1	CanTxDBG_760_E
EYE760_3_1_EGI145_MAC_NG_FLG	0	0	CanTxDBG_760_E
EYE760_2_0_EGI044_MAC_OK_FLG	1	1	CanTxDBG_760_E
EYE760_2_1_EGI044_MAC_NG_FLG	0	0	CanTxDBG_760_E
EYE760_1_0_MSG_COUNTER	14	E	ƒƒbƒZ[ƒWƒ]ƒEƒ`ƒ

Figure 14: When Receive Control Message is Not received.

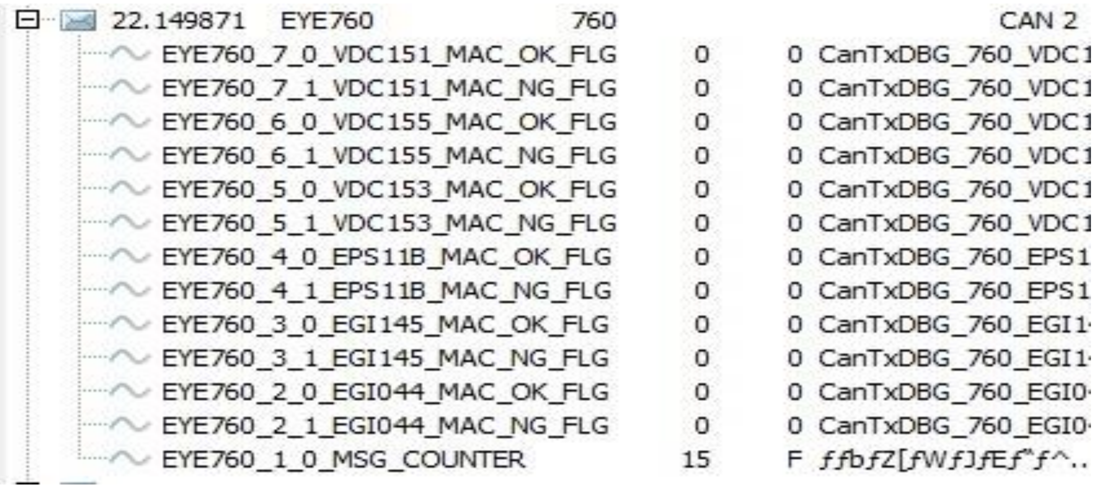
Even though Rx Control Message is received, MAC Verification is failed due to wrong entry of Key and MAC_NG Flag is set High, same can observed in the below Figure 15.



19.799026 EYE760	760		CAN 2
EYE760_7_0_VDC151_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_7_1_VDC151_MAC_NG_FLG	1	1	CanTxDBG_76
EYE760_6_0_VDC155_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_6_1_VDC155_MAC_NG_FLG	1	1	CanTxDBG_76
EYE760_5_0_VDC153_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_5_1_VDC153_MAC_NG_FLG	1	1	CanTxDBG_76
EYE760_4_0_EPS11B_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_4_1_EPS11B_MAC_NG_FLG	0	0	CanTxDBG_76
EYE760_3_0_EGI145_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_3_1_EGI145_MAC_NG_FLG	0	0	CanTxDBG_76
EYE760_2_0_EGI044_MAC_OK_FLG	0	0	CanTxDBG_76
EYE760_2_1_EGI044_MAC_NG_FLG	1	1	CanTxDBG_76
EYE760_1_0_MSG_COUNTER	6	6	ƒƒbƒZ[ƒWƒ]ƒ

Figure 15: MAC Verification Failed

Successful MAC verification is observed on the results when same key is used for both the sending and receiving ECU and there are no data changes while sending from sender to receiver ECU. Figures 16 shows the Successful MAC verification done at the receiver and it set MAC_OK flag is HIGH.



```
⊟ 22.149871   EYE760                        760                       CAN 2
     EYE760_7_0_VDC151_MAC_OK_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_7_1_VDC151_MAC_NG_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_6_0_VDC155_MAC_OK_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_6_1_VDC155_MAC_NG_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_5_0_VDC153_MAC_OK_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_5_1_VDC153_MAC_NG_FLG           0        0  CanTxDBG_760_VDC1
     EYE760_4_0_EPS11B_MAC_OK_FLG           0        0  CanTxDBG_760_EPS1
     EYE760_4_1_EPS11B_MAC_NG_FLG           0        0  CanTxDBG_760_EPS1
     EYE760_3_0_EGI145_MAC_OK_FLG           0        0  CanTxDBG_760_EGI1
     EYE760_3_1_EGI145_MAC_NG_FLG           0        0  CanTxDBG_760_EGI1
     EYE760_2_0_EGI044_MAC_OK_FLG           0        0  CanTxDBG_760_EGI0
     EYE760_2_1_EGI044_MAC_NG_FLG           0        0  CanTxDBG_760_EGI0
     EYE760_1_0_MSG_COUNTER                15        F  ƒƒbƒZ[ƒWƒ]ƒEƒ"ƒ^..
```

Figure 16: MAC Verification Successful.

VII. CONCLUSION

Current automotive architectures have not been designed with security in mind. The increasing amount of connectivity inside and between vehicles and the Internet in recent years made such approaches necessary to ensure the safety of passengers. While gateways to external networks are often protected, internal networks are seldom separated in terms of security. Authentication, authorization and encryption are typically not used and often cannot be employed due to the restriction of the underlying communication systems. due to the typical bus structure of internal networks, an attacker has full access to all functions, once he penetrated the external gateway. The influence across components is especially high in a bus structure, as all messages could be sent and received by all communication participants.

Especially the influence on safety is important, as it can lead to loss of life in the worst case, e.g. when a vehicle is under attack. A security mechanism exceeding the real-time requirements of the vehicle, can lead to similar consequences. While much work has been performed on securing external interfaces and connections with firewalls and gateway systems, the internal vehicle networks have not received the required attention.

Based on the above requirements, this paper proposes the design and implementation of message authentication framework, allowing to efficiently secure the internal communication. Message authentication corresponds to any content of the data payload created exclusively for ensuring the authenticity, security of the remaining information in the data payload. Calculation of the message authentication code normally depends on the Authentication Data, Secured Key, CMAC algorithm, and counter information, thus limiting the possibility of simulating a message by a hacker to the vehicle network.

The approaches to secure automotive architectures in this thesis can only form an initial step into the large area of automotive systems security. Further work is required to ensure the security of automotive architectures and their components. One of the major tasks, both in terms of time and effort, is the standardization of any proposed approach. Only through standardized approaches, it is possible to avoid custom, potentially insecure solutions and enable the clean integration of thoroughly tested approaches. Standardization further fosters widespread adoption of approaches. Thus, it forms the basis of future automotive security solutions.

This paper focuses on the security of the messages and networks in the architecture. Additional work is required in both analysis and design of ECUs. The approaches proposed in this thesis rely on the software on ECUs to be secure. This needs to be ensured with secure (key) storage, secure execution environments, secure boot, etc.

VIII. ACKNOWLEDGMENT

The authors will like to express sincere thanks and appreciation to his esteemed mentor and supervisor for their ongoing inspiration, technical support, guidance, encouragement, and insightful suggestions that enabled him to undertake this research work.

REFERENCES

[1] R. A. Kemmerer. Cybersecurity. In 25th International Conference on Software Engineering, 2003. Proceedings., pages 705–715, May 2003.

[2] SAE International. SAE J3061_201601 - Cybersecurity guidebook for cyber-physical vehicle systems, Jan. 2016.

[3] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, 1(1):11–33, 2004.

[4] D. G. Firesmith. Common concepts underlying safety security and survivability engineering. Technical Report CMU/SEI-2003-TN-033, Software Engineering Institute - Carnegie Mellon University, Dec 2003.

[5] J. Erjavec and R. Thompson. Automotive technology: a systems approach. Cengage Learning, 2014.

[6] P. Carsten, T. R. Andel, M. Yampolskiy, and J. T. McDonald. In-vehicle networks: Attacks, vulnerabilities, and proposed solutions. In Proceedings of the 10th Annual Cyber and Information Security Research Conference, pages 1–8. ACM, 2015.

[7] P. Kleberger and T. Olovsson. Protecting vehicles against unauthorised diagnostics sessions using trusted third parties. In F. Bitsch, J. Guiochet, and M. Kaâniche, editors, Computer Safety, Reliability, and Security: 32nd International Conference, SAFECOMP 2013, Toulouse, France, September 24-27, 2013. Proceedings, pages 70–81, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[8] P. Kleberger and T. Olovsson. Securing vehicle diagnostics in repair shops. In A. Bondavalli and F. Di Giandomenico, editors, Computer Safety, Reliability, and Security: 33rd International Conference, SAFECOMP 2014, Florence, Italy, September 10-12, 2014. Proceedings, pages 93–108. Springer International Publishing, 2014.

[9] P. Kleberger, T. Olovsson, and E. Jonsson. An in-depth analysis of the security of the connected repair shop. In The Seventh International Conference on Systems and Networks Communications (ICSNC), Proceedings. Lisbon, 18-23 November, 2012. IARIA., page 99, 2012.
[10] G. Dimitrakopoulos and P. Demestichas. Intelligent transportation systems. IEEE Vehicular Technology Magazine, 5(1):77–84, March 2010.
[11] D. K. Nilsson and U. E. Larson, "Simulated Attacks on CAN Buses: Vehicle virus," in Proceedings of the Fifth IASTED Asian Conference on Communication Systems and Networks (ASIACSN), 2008.
[12] D. K. Nilsson and U. E. Larson, "Simulated Attacks on CAN Buses: Vehicle virus," in Proceedings of the Fifth IASTED Asian Conference on Communication Systems and Networks (ASIACSN), 2008.
[13]D. K. Nilsson, U. E. Larson, F. Picasso, and E. Jonsson, "A First Simulation of Attacks in the Automotive Network Communications Protocol FlexRay," in Proceedings of the First International Workshop on Computational Intelligence in Security for Information Systems (CISIS), 2008
[14] M. Wolf, A. Weimerskirch, and C. Paar, "Security in Automotive Bus Systems," in Workshop on Embedded IT-Security in Cars, Bochum, Germany, November 2004.
[15]T. Hoppe and J. Dittman, "Sniffing/Replay Attacks on CAN Buses: A simulated attack on the electric window lift classified using an adapted CERT taxonomy," in Proceedings of the 2nd Workshop on Embedded Systems Security (WESS), Salzburg, Austria, 2007.
[16] A. Lang, J. Dittman, S. Kiltz, and T. Hoppe, "Future Perspectives: The car and its IP-address - A potential safety and security risk assessment," in The 26th International Conference on Computer Safety, Reliability and Security (SAFECOMP), Nuremberg, Germany, 2007.
[17] Groza and S. Murvay, "Efficient protocols for secure broadcast in controller area networks," IEEE Trans. Ind.
Informa., vol. 9, no. 4, pp. 2034–2042, Nov. 2013.
[18] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive CAN networks—Practical examples and selected short-term countermeasures," Rel. Eng. Syst.
[19] "Overview of Bosch Motronic Hardware," http://volvospeed.com/obd2/ ecu overview.htm, visited November, 2007.
[20] National Semiconductor, "CP3SP33 Connectivity Processor with Cache, DSP, and Bluetooth, USB, and Dual CAN Interfaces," Datasheet, 2005.
[21] Bosch, "CAN Specification 2.0," http://www.dcd.pl/dcdpdf/can2spec. pdf, 1991, visited August, 2007.
[22] M. Bellare, J. Kilian, and P. Rogaway, "The Security of the Cipher Block Chaining Message Authentication Code," Journal of Computer and System Sciences, vol. 61, no. 3, pp. 362–399, 2000.
[23] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES modes of op- eration," in Proceedings of the 38th Annual Symposium on Foundations of Computer Science, 1997, pp. 394–403.