Protection against attack D.o.S. in CAN and CAN-FD vehicle networks

Luiz Quintino, Alexei Machado

Electrical Engineering – Pontifícia Universidade Católica de Minas Gerais (PUC-MG)
Campus Coração Eucarístico
30.535-901 – Belo Horizonte – MG – Brasil

luiz.quintino@gmil.com, alexeimcmachado@gmail.com

Abstract. *For some time the vehicles adopted distributed electronic systems as a way to bring more comfort, active and passive safety to the occupants. With the increase of technological demand and innovations in connectivity, vehicles became attractive to hackers. Regardless of the goals of hacking, the future of automobiles goes through a wide range of related work evaluating the impacts of malicious people that take control of the vehicle's systems. In this article we evaluate the vulnerabilities of a vehicular data bus, CAN and CAN-FD. It is possible for an intruder to interfere with instrument panel display systems by sending forged messages to the driver and even compromise systems related to safety such as steering and brakes. Among the various types of possible attacks on the CAN bus, the D.o.S. (Denial of service) lacks of a deeper discussion in the literature. This article proposes a mechanism to protect the bus from this type of attack. A filter is designed to avoid that the bus be overloaded with false requests, providing access to authentic message exchanging.*

1. Introduction

The growth of automation in the vehicles and the connectivity with external devices allows modules to make decisions and take control of vehicular systems that controls comfort and safety (Larson, et al., 2008). Some systems control vehicle safety functions autonomously, without the driver intervention. Emergence Brake automatically reduces the vehicle's velocity in the presence of obstacles or pedestrians; Stop & Start shuts off the engine when the vehicle stops and restarts it automatically. These are some examples of features in modern vehicles related to the safety and comfort of the users. However, several studies have demonstrated weaknesses in the safety of electronic modules and current communication networks in vehicles (Larson, et al., 2008) (Wampler, et al., 2009) (Kleberger, et al., 2011).

Vehicles vulnerability becomes a point of attention and the hacker's interest on this type of technology is increasing. In a presentation on vehicle safety, Black Hat 2015, a hacker demonstrated vulnerabilities in a vehicle, in real time, remotely accessing their systems and gaining control over some modules. The involved vehicle manufacturer corrected the vulnerabilities, generating a recall of 1.4 million vehicles in the USA (Miller, et al., 2015). More recently Tesla and VW vehicles had a similar

problem and again, the modules needed to be re-flashed to protect the vehicle against malicious attack.

There is several kinds of attack that could be performed in a vehicular network. In this work we propose to minimize the effect of Denied of Service (D.o.S.) attack on any type of network that uses Controller Area Network (CAN) and CAN-Flexible Data-rate (CAN-FD) buses. The mechanism uses the arbitration layer of the transceiver to control the messages, avoiding the bus overload that occurs during a D.o.S. attack. Several tests were simulated on a bus mounted on a test bench to evaluate different conditions of attack. The goal is to have a minimum loss of authentic messages on the bus in the face of a D.o.S. attack in order to guarantee the correct operation of critical and security-related systems.

2. Vehicle network protection

Network protection has been studied extensively for communication between computers. The need for study came with the spread of Internet use to a growing number of purposes. Vehicle networks have more recent studies, mainly because there was no interest in attacking this kind of system. With the increase of use, the network communication in vehicles and the recent possibility of connecting the vehicle with the outside world, using cellphones connectivity, the vehicle became attractive and safety studies for this type of network tends to increase.

2.1 Types of vehicular network

The networks used in vehicles are presented in Figure 1. CAN is even the most used. Local Interconnected Network (LIN) is used for low cost systems complementing the CAN buses, used for sensors and no complex modules, Most and FlexRay are used for critical systems in luxury vehicles. There are studies to use Ethernet in the near future (Yong Kim, 2011).

Bus network	Characteristics	Bus type	Speed (Kbps)
LIN	Universal Asynchronous Receiver / Transmitter (UART)	Star; one wire	20
CAN	Carrier Sense Multiple Access (CSMA/CR)	Ring or star; twisted pair	1.000
FlexRay	Time Division Multiple Access (TDMA)	Star; Twisted pair	10.000
MOST	TDMA synchronous	Ring; Twisted pair or optical fiber	50.000 (optical fiber) 150.000
Ethernet	Commutation Full-Duplex	Star; Twisted pair or dual Twisted pair	100.000

Figure 1 - Characteristics of busbar used in vehicles (Yong Kim, 2011)

In the example of Figure 2, a typical vehicular architecture is presented with two independent CAN buses (CAN-1 and CAN-2). The buses are interconnected through a Bridge or Gateway. In this example one of the Electronic Control Unit (ECU5) has several external communication interfaces (internet, bluetooth, WiFi, etc.), a physical

connection to the bus, anda connector called On Board Diagnosis II (OBD-II) used to diagnose communication with external equipment connected to the bus.

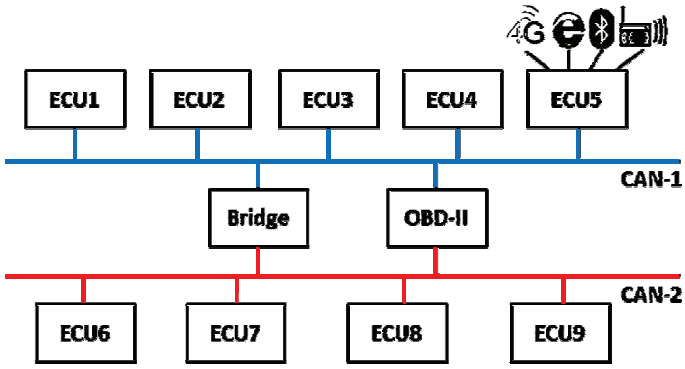


Figure 2 - Example of vehicular architecture (Wang, et al., 2014)

2.2 Characteristics of a CAN Network

The CAN network is based on the OSI model and has a layered structure where the lower layer is responsible for accessing the bus. One of the main features of the CAN network is to avoid collision, that means, there is no collision of messages on the physical bus.

A CAN message (Figure 3) is composed of an identifier (ID), control bits, 8 bytes of data, a Cyclic Redundancy Check (CRC), which is a mathematical calculation to test the integrity of a message, a bit of Acknowledgment (ACK) that is nothing more than a confirmation from the other nodes on the reception of a transmitted message, and a few bits reserved for error control in the End Of Frame (EOF) (ISO, 1999).

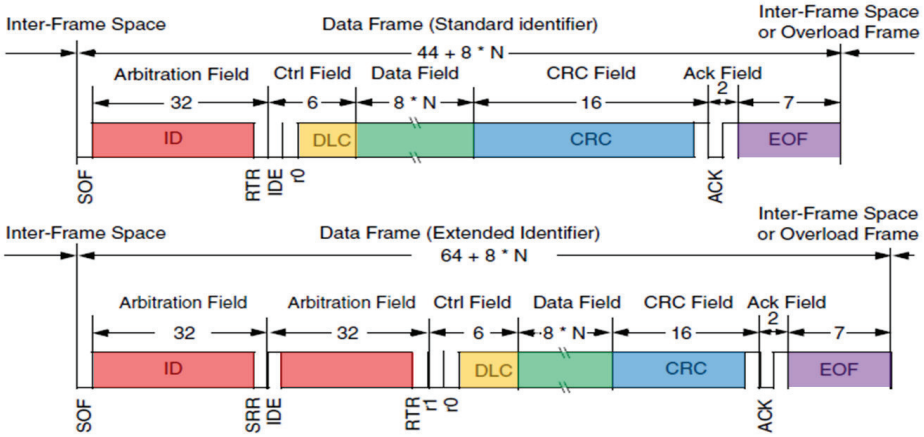


Figure 3 - Date frame of a message CAN (ISO, 1999)

The purpose of the identifier is to identify the message on the network. In this way, each different message has a unique ID that determines what that message is. As in the CAN network there is no information about the origin and destination of the message, it is the ID that will orienting the node that receives the message to filter and decide if each message is useful or not.

In addition to identifying the message, the ID also performs an important function of determining the priority of access to the CAN bus. The lower the ID number of a message, the higher its priority. In the case that two or more nodes want to transmit at the same time on the bus the ID is the first information transmitted to the bus. Each time a bit of the ID is transmitted this bit is read back by the sender node. Since bit 0 is dominant and bit 1 is recessive, smaller IDs will be dominant on the bus. When a node is transmitting but receives a dominant bit, the transmission is ceased, allowing the higher priority message to be transmitted. The node or nodes that fail to transmit will wait until the end of the next EOF, at which point the waiting nodes gain access to the bus again. Figure 4 shows a diagram representing two nodes, A and B trying to transmit at the same time. It can be observed that node A obtained access to the bus every time it wrote messages with the lowest ID value.

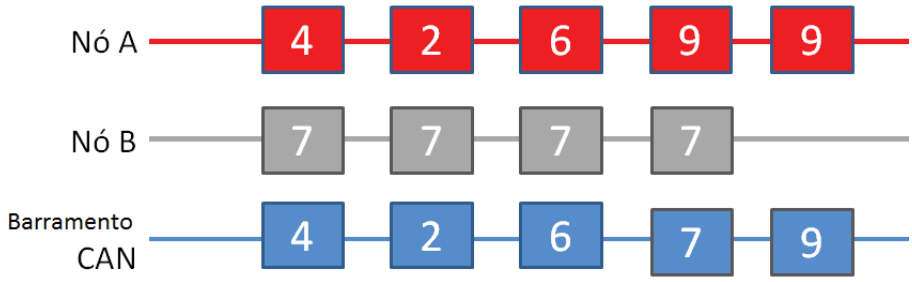


Figure 4 - Access arbitration to the CAN bus, adapted from Corrigan (Corrigan, 2008)

A CAN message may have IDs of 11 or 29 bits in length. The same bus may contain nodes that work with IDs of different sizes, as long as the node that will receive the message be able to identify it. A node that can only read 11 bit IDs may not receive a 29 bit message, the Identification Extended (IDE) bit is used to identify when a 29 bit ID is being used in the message.

The CAN bus has to work with a busload or “bus utilization rate” bellow 60%, to guarantee gaps in the bus so that all the modules can access it (Natale, 2008). In the development of the network busload, it must be calculated to evaluate the need to add new buses and to accommodate the amount of messages.

The CAN-FD uses the same principles of conventional CAN, but with greater speed and number of bytes. While the conventional CAN network can transmit at most 8 bytes of data at 1Mbps, the CAN-FD can transmit up to 64 bytes of data up to 12Mbps. For achieving a compatibility between CAN and CAN-FD, the same functional concept used in the CAN network was adopted in CAN-FD, and the high speed only happens in the data field phase, hence the name FD which means Flexible Data-rate, that is, it is possible to modify the transmission rate flexibly in the middle of the transmission.

2.3 Attacks and protection of vehicular networks

Nilson (Nilsson, et al., 1996) uses some criteria to classify and measure attack risks that make it possible to modify the firmware of electronic modules. Classification is based on Safety Integrity Level (SIL). Safety integrity is the likelihood of a safety-related system performing satisfactorily the required safety functions under all conditions in a given period of time. The Safety effect Level (SEL) classifies the effects of a malfunction on systems in levels 1 to 4 (Figure 5 - SEL according to the effect of the security threat).

Safety effect	Safety effect level (SEL)
Disastrous	4
Severe	3
Mediocre	2
Distracting	1

Figure 5 - SEL according to the effect of the security threat (Nilsson, et al., 1996)

In the work of Wolf (Wolf et al., 2004) and Nilson (Nilsson, et al., 2008) the weaknesses of the current networks are highlighted, since studies show protection in enough vehicular networks to allow malicious attacks and the possibility of success in creating such attacks.

Mahmud (Mahmud et al., 2006) states that a series of analysis should be carried out to study and balance the risks and costs involved in implementing a complete security policy, such as: classifying vehicle systems for risks inherent in its functionality as loss of comfort, loss of performance, accident risk, classifying critical modules, analyzing messages and signals that must be protected, analyzing impacts related to security breach in each class, and studying contingency plans.

Onishi (Onishi, 2012) estimates based on data from the US and Canada that damage from cyberattack has a potential loss of $ 56 million per year for every 1% of vehicles affected, considering the sales of a vehicle model. Onishi (Onishi, 2014) points out that vehicle system vulnerabilities are more complicated with respect to cyber-attacks than when compared to computers and the internet. Personal computers have about 2,000 components while vehicles have more than 20,000 components and up to 100 electronic modules with hundreds of megabytes of firmware code. Smit (Smith, 2016) presents techniques for gaining access to a vehicle's modules and networks, from hardware recognition, networking, firmware, electronics, and even to circumventing systems, entering commands, and modifying firmware's.

3 D.o.S. attack

The D.o.S. attack consists of flooding a network with messages in order to prevent its normal traffic (Nilsson, et al., 2008). It is important to know the D.o.S. concept in the computers to understand the mechanism used in the vehicle network. This prior knowledge is important, even knowing in advance that there is a difference in the network architecture and Internet protocol when compared to the CAN network. Likewise, solutions presented for one type of network do not apply directly to the other.

It is verified that there is no effective solution that guarantees 100% of protection for an Internet network against D.o.S. attacks and no solution has been found in the literature to protect a CAN network against a D.o.S. attack.

The consequences of a DoS attack on a vehicle are as devastating as the level of technology available in the vehicle. Decentralization of functions requires that systems communicate to make decisions. This means that a D.o.S. attack on a CAN network will completely stop any type of communication and put in risk the systems decisions. The more distributed the systems is and dependent on sensors and information that travel through the CAN bus, the bigger will be the consequences.

3.1 D.o.S. attack simulationon CAN network

A simulation environment composed of an architecture formed by several interconnected nodes was implemented for the experiments. The bus was configured to operate at 125kbps. The CANoe tool was used to create a message map with dozens of messages of different sizes and ID's. A message traffic flow was created in the message map between the nodes, with the definition of the nodes that will transmit each message and the periodicity thereof. The CANoe is a Vector tool designed to simulate vehicle buses. Vector is the biggest supplier of tools to monitor, controls, diagnostic and simulate vehicle buses. CANoe is a professional tool used from component developers and could simulate all kinds of vehicle busses other than CAN.

Figure 6a shows an image from an oscilloscope presenting a large empty space between messages on the bus. At this point the busload is about 14%. It is recommended that a CAN messaging architecture be designed to work with up to 60% busload, ensuring enough spaces for all packages to be delivered with minimal delay (Corrigan, 2008). The arbitration mechanism of the CAN network guarantees that messages with a smaller ID will have priority access and will guarantee less delays in relation to less priority messages. When a CAN network is under attack there is no space between messages (Figure 6b). The attacker uses the lowest ID preventing authentic messages from being transmitted.

Figure 6 – a) Normal CAN bus 14% bus load; b) CAN bus under attack 100% bus load
Source: from author

3.2 Attack protection for D.o.S. attack over a CAN and CAN-FD network

A way to mitigate D.o.S. attacks on a CAN bus could be to limit the number of messages activing the bus from the same node in a period of time. As this type of control is not possible in a centralized way, an alternative was designed for each node to have autonomy to manage a control, filtering the message before sending it.

Since each node is responsible for managing the access to the bus and each node needs a transceiver, the method presented in this paper uses the CAN transceiver to control the message flow during an attack. The idea is based on a constant flow control using a hardware filter implemented in the transceiver with the capability to limit the access to the bus. Since every node has a transceiver, it is guaranteed that access to the bus will be controlled. The filter is able to limit the number of messages that a node will transmit within a period of time, allowing the other nodes to also have access to the bus.

Each time a message arrives from the micro-controller to be sent, the transceiver must check the value of an internal counter. If the counter is less than a value set as a configurable filter parameter the message is transmitted and the counter is increased. If the counter is greater than the set value, the message will not be sent and the node will wait until the counter value is decremented, allowing a limited number of consecutive messages to be sent. The way to decrement the counter is related to a timer configured to count a time t, which is also a configurable parameter. Each time the timer reaches the set time it will decrement the counter until it reaches zero, so the timer will generate a delay in the transmission of the message during an attack.

This mechanism allows messages that are transmitted within normal communication intervals, which generate busloads below 60%, do not have their transmission altered by the filter, and in this way there will be no packet loss or transmission delays.

In case of a D.o.S. attack, the node promoting the attack will have its messages controlled by the filter and after sending a sequence of consecutive messages within the established time limit, it must wait for a delay before being able to access it again. This waiting time allows the other nodes to compete for access to the bus and thus the main functions of the vehicle will be performed even during an attack.

The filter algorithm could be incorporated into both a CAN transceiver and a CAN-FD. The implementation can be done by software but it would be more indicated by hardware, for hindering the possibility of adulteration of the configurations.

3.3 Filter parameterization

The configuration values for the number of consecutive messages sent and time to decrement the counter was evaluated in order to find an ideal value for maximum efficiency and minimum loss of packages. This generates a delay in the next sending of messages by the node. In this way it would be possible for the transceiver to be programmed from the factory with a default setting value that cannot be changed. The measurements in the simulator have the objective of finding the optimal values for the parameters that represent the minimum of packet loss with the minimum influence on the normal communication of the bus.

During the tests, two types of information were collected: the busload of the network and loss of message packs. The busload rate is important to analyze the efficiency of the filter in keeping the bus in operational condition, seeking not to change the flow of messages compared to normal operation. Similarly, busload information will also help to analyze network conditions during an attack and the influence of the filter in attempting to establish a control in the message flow.

4. Experiments

Initially the CAN network was configured to 125kps, with a busload of 10%. The number of messages or packets transmitted under normal conditions was measured. Five measurements were taken to register the average of authentic messages transmitted within 1 second. It was observed that some messages have a higher periodicity than others, and therefore are transmitted more often.

An attack was introduced using the 0x90 ID, as being an ID smaller than the lowest ID used in the messages that were originally being transmitted (Figure 7). The time between the attack messages was been set in the 960µs range, since a CAN message at a rate of 125kbps has this duration time with 8 byte of Data Length Control (DLC). This attack generated a 100% busload and a loss of 100% of the authentic messages, completely disabling the bus. In this way, no node could communicate with others and services that depended on messages, commands or information from other nodes would be unavailable. It was also noted that the number of packets or messages sent was about 993 messages, rather than 121 in normal operation, which means the maximum number of queued messages that can travel on the bus in the period of 1 second.

To demonstrate the influence of the periodicity on the message identifier, an attack test was performed with the ID 0x1E40000 that is bigger than the authentic ID's used. Despite the busload having reached 100% saturation, there was no packet loss and all 121 authentic messages were successfully transmitted without delays, since all authentic IDs have priority over the attacker ID.

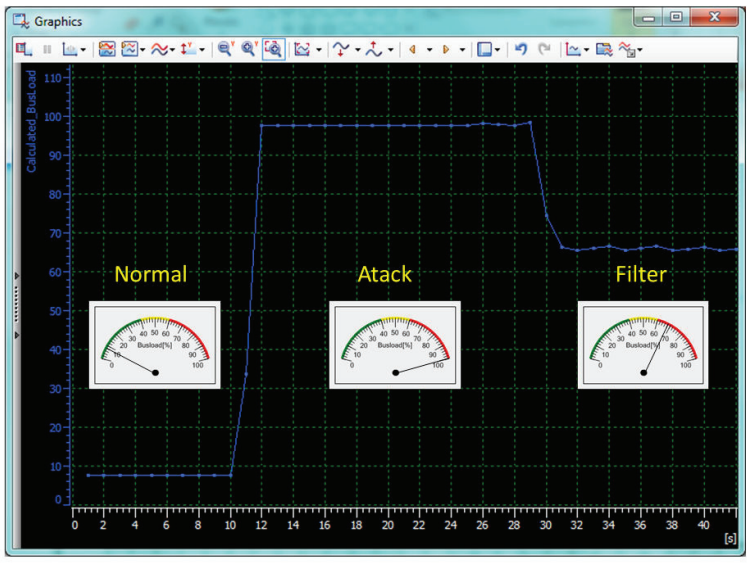
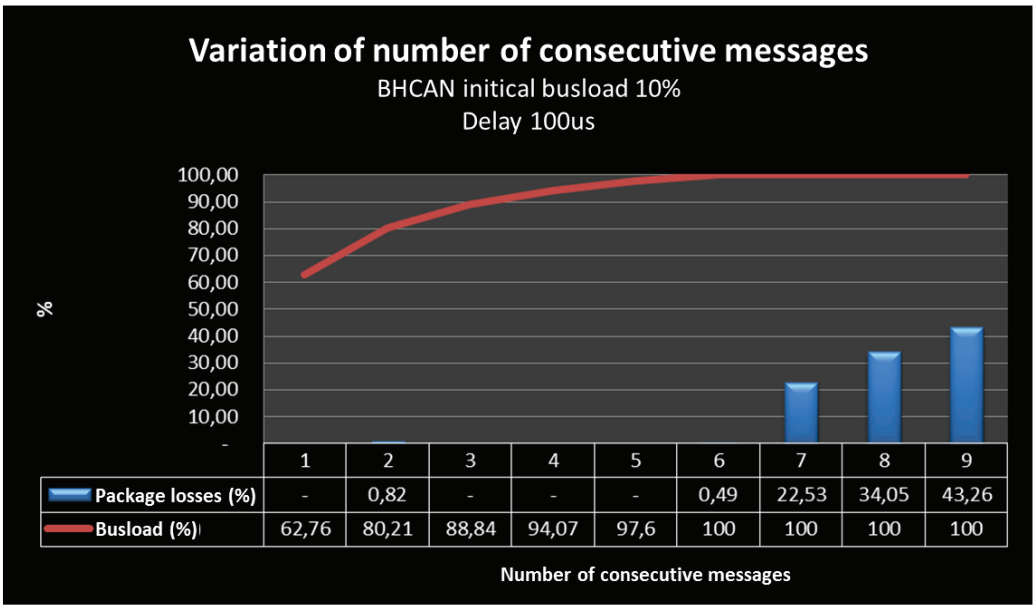


Figure 7 - Busload view during filter test
Source: from author

A test was also made considering the CAN bus with 125kbps and initial busload of 10%, with message delay of 100µs and varying the number of consecutive messages from 1 to 9. The result is shown in Figure 8. It can be observed that the number of consecutive messages has a direct influence on busload and packet loss during an attack. A higher number of consecutive messages that the node sends will generate a higher busload rate during an attack and the greater the number of packets lost.



	1	2	3	4	5	6	7	8	9
Package losses (%)	-	0,82	-	-	-	0,49	22,53	34,05	43,26
Busload (%)	62,76	80,21	88,84	94,07	97,6	100	100	100	100

Figure 8 - Filter test with variation of the number of consecutive message loss of packs (%) and busload rate (%) in function of the variation of the number of consecutive messages allowed by the filter varying from 1 to 9.

Source: from author

4.1 Message filter settings

In order to obtain an optimum calibration value of the filter parameters, several conditions were tested, varying the initial busload, waiting time, number of consecutive messages and bus speed. In total, 21 delay times, 9 consecutive message variations for 3 initial busload variations were made for CAN at 125kbps. These measurements were repeated for CAN at 500kbps with 13 time delay variations, 9 consecutive message variations for 2 initial busload variations, repeating the measurements 5 times to obtain the average of packages, making a total of 4005 measurements.

In order to better represent the values found in the tests, the graph of Figure 9 was plotted. The vertical axis shows the busload rate as a function of the delay time in the attack messages. Each curve represents the variation of the number of consecutive messages. The variation in delay time was started with a range of 100µs, representing 10% of the message size. As can be observed this variation only had an effect on the busload rate after a delay of 1000µs, which represents the 100% of message length. Due to this measured characteristic was used a new interval of 500µs in the delay time, that is, 50% of the message length.

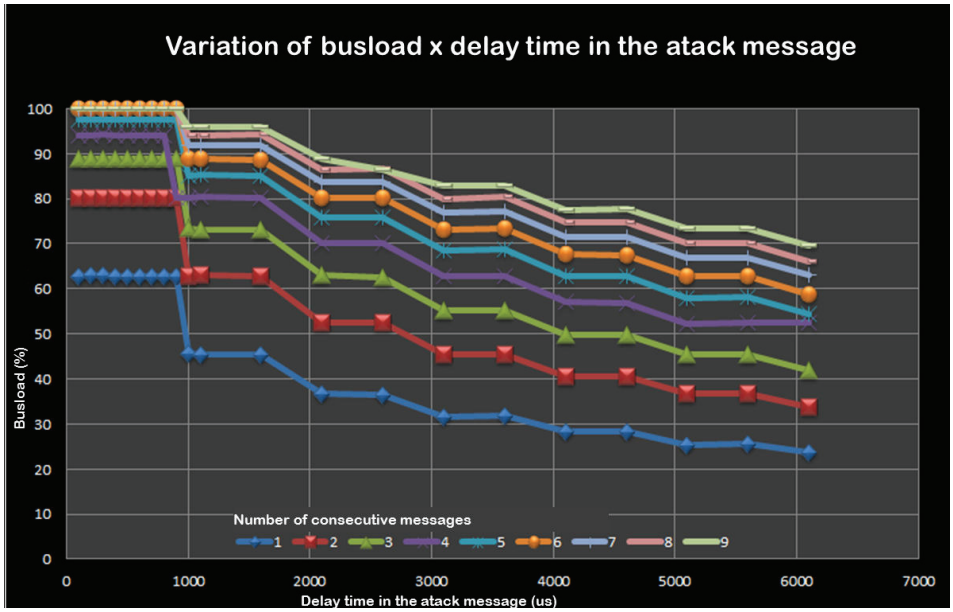


Figure 9 - Busload measurement during D.o.S. attack with the filter active
Source: from author

The tests were made for CAN with initial busload of 20% and 60% at 125kbps and at 500kbps. It can be observed (Figure 10) that the margin for the filter to act with the initial busload at 60% is very small, but it was still possible to find calibration values that kept the busload below 100% during an attack. In the same way, with 60% of initial saturation there were losses of packages that reached 100% for a certain range of calibration of the parameters, however, the filter was still effective in the green area highlighted in the graph, in which there was no loss of package (Figure 11).

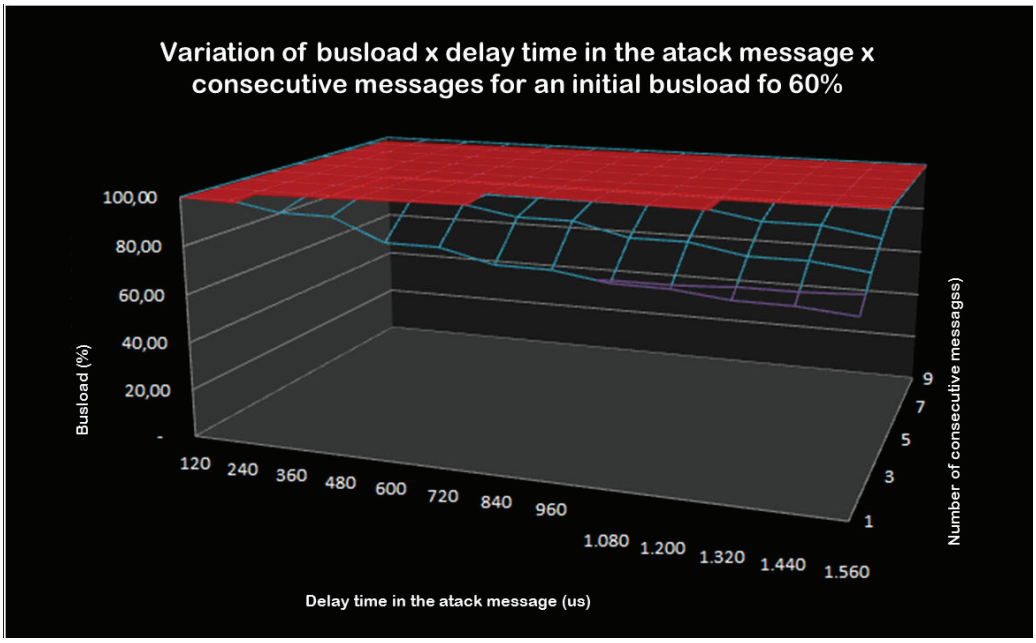


Figure 10 - Busload for CAN at 500kbps with initial busload of 60%
Source: from author

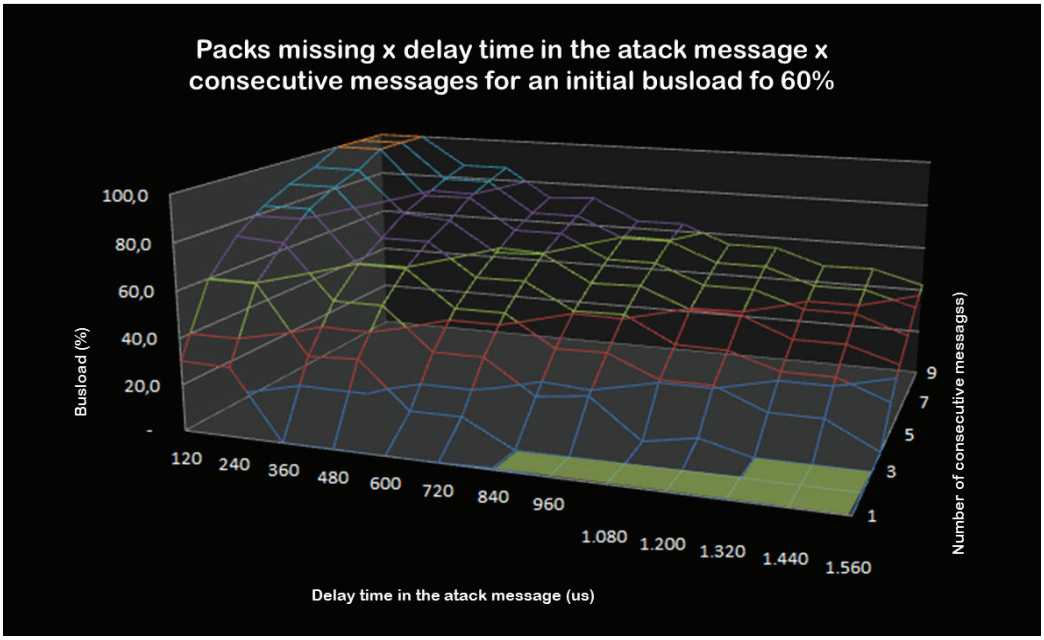


Figure 11 - Messages losses for CAN at 500kbps with initial busload of 60%
Source: from author

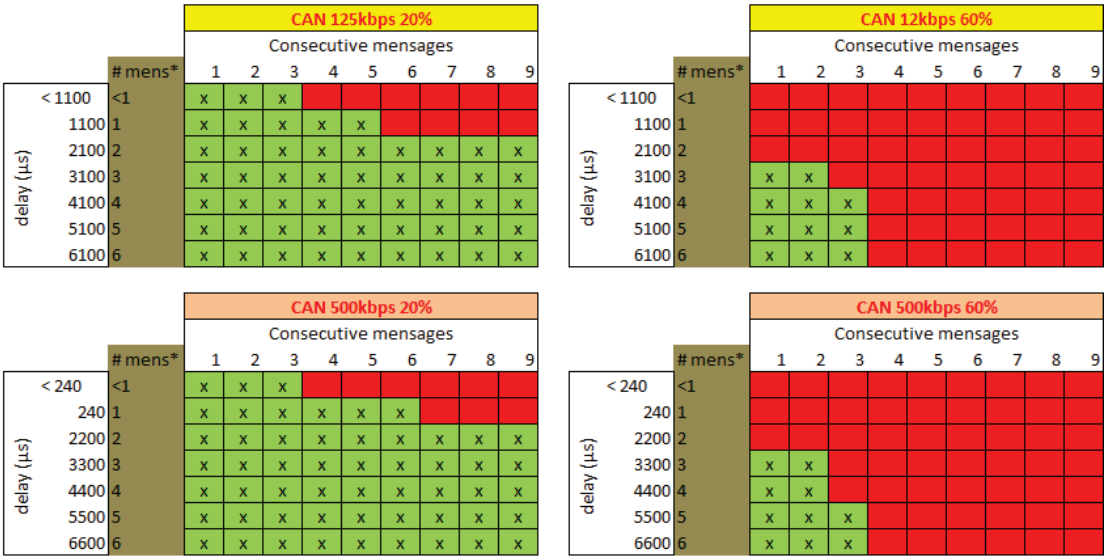
4.2 Analysis of results

The performed measurements indicate that the filter can be efficient even in the most critical condition of busload, 60%, and with a baud rate of 500kbps in all test conditions with the proper calibration adjustments. It was noted that the initial bus saturation and the baud rate significantly modified the effectiveness of the filter. Figure 12a shows all condition tested. The green area is where the filter was efficient in dealing with the D.o.S. attack. The Figure 12b summarizes the filter settings overlapping. It may be noted that there is a region where the filter was able to deal with the attack under all conditions of the bus. This means that the yellow region values in the frame can be used for all bus conditions tested. In the table, the delay time was indicated in relation to the size of a message, in order to refer to any bus speed. As was demonstrated, the size of a message on at 125kbps bus is 970µs and at 500kbps it is 240µs.

A single node can send several different messages, with different periodicities and IDs. Are the parameters found for the filter enough to guarantee that in normal condition the messages will not be delayed? There are two ways to analyze this question. The first one is regarding to regular messages; the second one, for diagnostic service messages.

For regular messages in a CAN bus with 125kbps, the modules use to have periodicity between 100ms and 2.000ms. In general one module sends from one to 5 different messages. Consider the message length equal to 1ms and the filter adjusted to 3 consecutive messages with a delay of time of 5ms on average. Considering the worst case within 5 messages transmitted in 100ms of periodicity. The total time to transmit 5 messages will be 3x1ms + 5x1ms + 2x1ms. This means that after three messages the module will wait 5ms to send the other 2 messages in a total time of 10ms. For 100ms of periodicity the module will send these messages again after elapsed 90ms. This

means that in this worse case, the delay time from the filter is not significant. To simplify, *total_time* = (**round**((*<number_of_messages>* + 1) / 3) * 5 + *<number_of_messages>*) * 1ms. Even if we consider the module sending double of regular messages using the maximum period, this means all messages will be sent in a total time of (**round**((10 + 1) / 3) * 5 + 10) * 1ms = 25ms << 100ms. In a real situation the vehicle uses to have few numbers of high periodic messages comparing with low periodic messages.



(*)mens = number equivalent to the messages lenght, deppending of the CAN bus speed.

Figure 12a – Filter configuration map
Source: from author

Figure 12b – Filter configuration map
Source: from author

Diagnostic messages can reach up to 255bytes and there are firmware upload services that have multiple kbytes of data. Transmitting this kind of messages the filter will generate a considerable over time in transmission. If we consider the effect of the filter we need add the time of 5 messages for every 3 messages really sent. A diagnostic message could send 7 bytes of data per each message, because of the Transport Protocol implemented over CAN, to transmit big data. For a 2MB file, for example, divided per 7 bytes we would have 285,715 diagnostic messages necessary. Appling the formula, the total time to this transmission with the filter working is 761.905ms or 12.32 minutes.

Without filter the total time is 4,76 minutes. This means 259% over the original time. If we consider that there are currently firmware upload processes that take 1h20m, the filter will increase the total time to 3h30m, which would be unacceptable.

The diagnostic messages always have a low priority ID, typically over 0x700 for 11 bit ID's. As has been shown previously diagnostic messages do not represent a risk for bus attack. A simple way to solve diagnostic message transmission is to bypass the diagnostic messages in the filter without compromise the protection against D.o.S. attack.

5. Conclusion

In this work, a method was presented to minimize D.o.S. attacks on any type of network that uses CAN and CAN-FD buses by creating a filter in the arbitration layer to measure the messages, avoiding bus overload and packet loss which occurs during a D.o.S. attack. The results of tests made in a simulator confronted with information from a real network were presented.

The goals of having, during a D.o.S. attack, a minimum loss of authentic communications and do not fail of critical or security-related systems, was overpassed and the proposed filter could be parametrized to avoid completely the losses of authentic messages at the buss. The filter is able to limit traffic overhead, allowing authentic messages to be transmitted between nodes even during an attack.

References

Corrigan, Steve. 2008. Introduction to the Controller Area Network - CAN. *Texas Instruments - Application Report.* July de 2008.

ISO. 1996. ISO 11989 - CAN Network. *CAN Network.* 1996.

Kleberger, Pierre, Olovsson, Tomas e Jonsson, Erland. 2011. Security aspects of the in-vehicle network in the connected car. *IEEE - Intelligent Vehicles Symposium IV*. 5-9 de Junho de 2011.

Larson, Ulf E., Nilsson, Dennis K. e Jonsson, Erland. 2008. An Approach to Specification-based Attack Detection for In-Vehicle Networks. *2008 IEEE Intelligent Vehicles Symposium.* 4-6 de Junnho de 2008.

Mahmud, Syed Masud e Shanker, Shobhit. 2006. In-Vehicle Wiereless Personal Area Network (SWPAN). *IEEE Transactions on vehicular technology.* maio de 2006, Vol. 55, 3.

Miller, Dr. Charlie e Valasek, Chris. 2015. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA.* Agosto de 2015.

Natale, Marco di. 2008. Understanding and using the Controller Area Network. 30 de October de 2008, p. 47.

Nilsson, D. K. e Larson, U. E. 2008. Conducting forensic investigations. *First International Conference on Forensic Applications and Tech-*. 21-23 de janeiro de 2008.

—. 2008. Simulated attacks on CAN buses:. *5th IASTED Asian Conference on Communication Systems and Networks.* 2-4 de Abril de 2008.

Nilsson, Dennis K., Phung, Phu H. e Larson, Ulf E. 1996. Vehicle ECU classification based on safety-security characteristics. *Chalmers University of Technology.* 1996.

Onishi, Hiro. 2014. Approaches for Vehicle Cyber Security. 2014.

—. 2012. Paradigm change of vehicle cyber security. 2012.

Smith, Craig. 2016. *The Car hacker's handbook - A guide for penetration tester.* 2016.

Wampler, David e Fu, Huirong. 2009. Security Threats and Countermeasures for Intra-Vehicle Networks. *IEEE Computer Society - Fifth Inernational Conference of Inforation Assurance and Security.* 2009.

Wolf, M., Weimerskirch, A. e Paar, C. 2004. Security in Automotive Bus. *Workshop on Embedded IT-Security in Cars.* 11-12 de novembro de 2004.

Yong Kim, Masa Nakamura. 2011. Automotive Ethernet Network Requirements. *IEEE 802.1 AVB Task Force Meeting.* Março de 2011.