Controller Area Network attacks and defense mechanisms: Survey Created by

Preprint · June 2020

DOI: 10.13140/RG.2.2.25052.82561

1 author:

Omar Alkhateeb
Technische Universität Berlin
1 PUBLICATION **0** CITATIONS

SEE PROFILE

Controller Area Network attacks and defense mechanisms: Survey

Created by
Omar ALKHATEEB
Omar.alkhateeb@win.tu-berlin.de

Table of Contents

Figures

Introduction

In the 1980s, Bosch developed the CAN bus for networking control units. The aim of the development was to reduce the wiring harnesses in order to save costs and weight. The protection of bus systems against external manipulation was not a development goal at the time, so that no corresponding security mechanisms were provided. The topic of cyber security did not yet exist at the time.
Against the background of automated functions and / or vehicles, this topic is becoming increasingly important. Unprotected or insufficiently protected systems can be manipulated and thus pose a potential risk to vehicle occupants and the environment.
Nowadays everyone can get the technical equipment and knowledge using the internet. Special expertise to hack the CAN bus system is also no longer required; instructions can be found in corresponding forums.
It is therefore more important than ever to address countermeasures that help identify and respond to such cyber-attacks.

Keywords:
CAN: Controller Area Network; ECU: Electrical Control Unit; Intrusion Detection System; Intrusion Reaction System

Controller area Network(CAN)

CAN Bus is working with a broadcast techniques which means that every node can send and all nodes can receive. To decide which node is allowed to send in a specific time when multiple nodes want to send, CAN Bus uses Message Identifier(11 bit) which gives the Message with the lower Message ID the ability to control the BUS and sends its data. Each node (Electrical Control Unit) has a receive filter which checks if the message that is transferred on the bus is important for it or not.

Each ECU (Electrical Control Unit) consists of Microcontroller, CAN Controller, and Transceiver(connects the ECU to the CAN Bus). ECU can send four CAN frame types: (https://en.wikipedia.org/wiki/CAN_bus)

- Data frame: a frame containing node data for transmission
- Remote frame: a frame requesting the transmission of a specific identifier
- Error frame: a frame transmitted by any node detecting an error
- Overload frame: a frame to inject a delay between data or remote frame

Note: Bit on CAN Bus is represented by 1 for recessive and 0 for dominant.

CAN Message consists of : Start-of-frame(1 Bit), Identifier(11 Bits), Remote transmission request(1 Bit), Identifier extension Bit(1 Bit), Reserved(Bit 1), data length code(4 Bits), data field(0-64 Bits), CRC (15 Bits) CRC delimiter (1 Bit must be recessive), ACK slot (1 Bit), ACK delimiter (1 Bit must be recessive) and End-of-frame (7 Bits must be recessive).

There are two error counters in CAN:

1. Transmit error counter (TEC)
2. Receive error counter (REC)

CAN Controller inside ECU has three states[1]:

- Error active(ECU sends active error frame)
- Error passive(ECU sends passive error frame)
- Bus-off(ECU stops sending)

When TEC or REC is greater than 127 and lesser than 255, a Passive Error frame will be transmitted on the bus.
When TEC and REC is lesser than 128, an Active Error frame will be transmitted on the bus.
When TEC is greater than 255, then the node enters into Bus Off state, where no frames will be transmitted.

[1] https://en.wikipedia.org/wiki/CAN_bus#Error_frame

Vulnerability in CAN Bus

The main problem in CAN is that the messages do not include any information about the source of the message which gives the attacker the ability to spoof messages, overload the bus with high priority messages or lead ECUs to Bus off state without knowing the attacker's identity.

Shawn Hartzell and Christopher Stubel [1] explain vulnerabilities in CAN in detail in their paper. Vulnerabilities are:

- Multicast messaging: every node can listen to the messages transferred on the Bus.
- Lack of Authentication: there is no way to be sure that the sender of the message is the supposed node.
- Lack of addressing: no source address in the message.
- Common Point of Entry: reaching the CAN bus is enough to reach all the nodes connected to it.
- Limited Bandwidth: make a challenge to add any cryptographic solution.
- Lack of Encryption: messages sent between nodes are plain.
- Multi-system integration

How to connect to CAN Bus

There is many ECUs connected to CAN Bus and some of them has a wireless or cellular connection with the world around the car. Checkoway et al [2] have described ways that an attacker can use to connect to CAN Bus by exploiting different access points:

- Indirect physical access(ex: OBD-II)
- Short-range wireless access(ex: Bluetooth)
- Long-range wireless access(ex: Broadcast channels)

What are the possible attacks

Attacks on CAN Bus can be categorized in different manner.
In my article I categorize them according to, if the ignition is ON or OFF and if the attacker uses Voltage-based attacks, Message-based attacks or malicious-code-based attacks.

Ignition OFF

Battery Drain Attack [3]:

the attacker exploits the wake up function in vehicle networks. by mounting the battery-drain attack, the adversary can increase the average battery consumption by at least 12.57x, drain the car battery within a few hours or days, and therefore immobilize/cripple the vehicle. Prerequisite: analyze how operation (e.g., normal, sleep, listen) modes of

ECUs are defined in various in-vehicle network standards and how they are implemented in the real world.

Denial of Body Control attack: [3]

The attacker exploits the wake up function in in-vehicle networks. The attacker can cut off communications between the vehicle and the driver's key fob by indefinitely shutting down an ECU, thus making the driver unable to start and/or even enter the car.
Prerequisite: analyze how operation (e.g., normal, sleep, listen) modes of ECUs are defined in various in-vehicle network standards and how they are implemented in the real world.

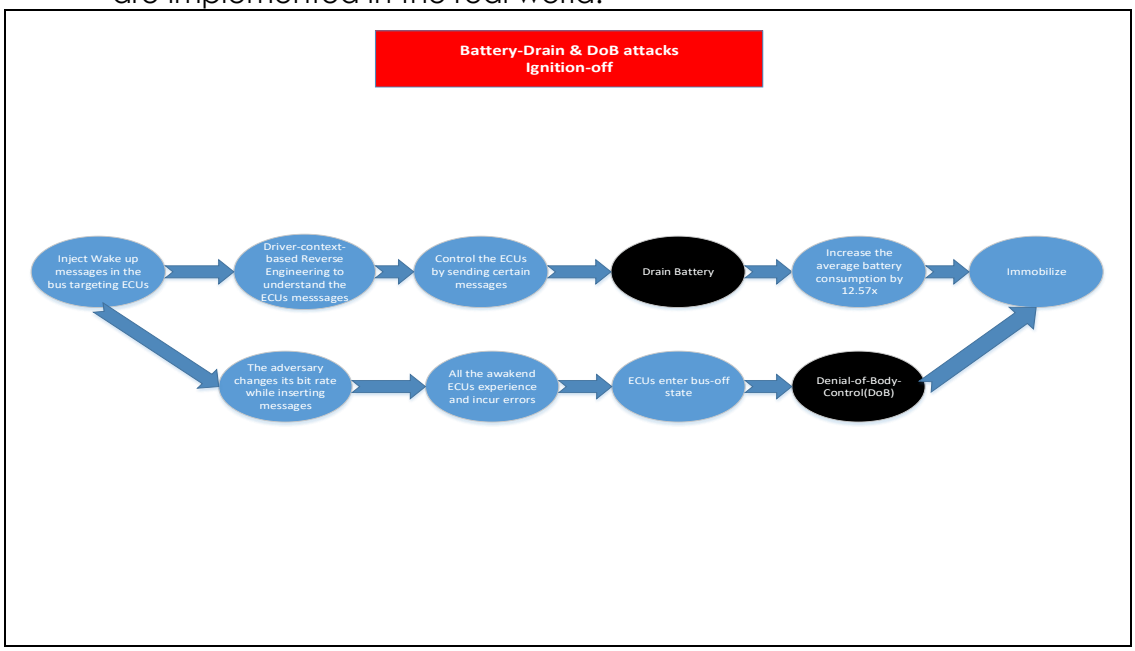

Figure 1: Battery-drain and DoB attacks

Ignition ON

Voltage-based attacks[2]

Forced Retransmission [4]

the adversary forces an ECU to retransmit by inducing an error during message exchange. The idea of this attack is making the transition time from a dominant bit to a recessive bit longer than the nominal transition time. By doing so, the adversary can induce an error in the message reception, especially at the ACK delimiter position which has to be the recessive bit.
Prerequisite: Knowledge of the CAN network voltage behavior.

[2] To understand this point you need to read the following https://en.wikipedia.org/wiki/CAN_bus#Physical

Denial Of Service(DOS on VIDS) [4]:

The idea of this attack is to increase the voltage level of CANL such that the differential voltage VDiff between CANH and CANL drops below the decision threshold for determining the dominant bit. Hence, the CAN bus observes a recessive bit while an ECU tries to transmit a dominant bit, and messages cannot be transmitted through the CAN bus.
Prerequisite: Knowledge of the CAN network voltage behavior.

Overcurrent Attack [4]:

The idea of this attack is to make the current that flows into the analog pin of the microcontroller exceed the absolute maximum rating of the microprocessor. thus rendering the microprocessor to malfunction or get burned due to the electric shock.
Prerequisite: knowing the current limit that the microprocessor can absorb.

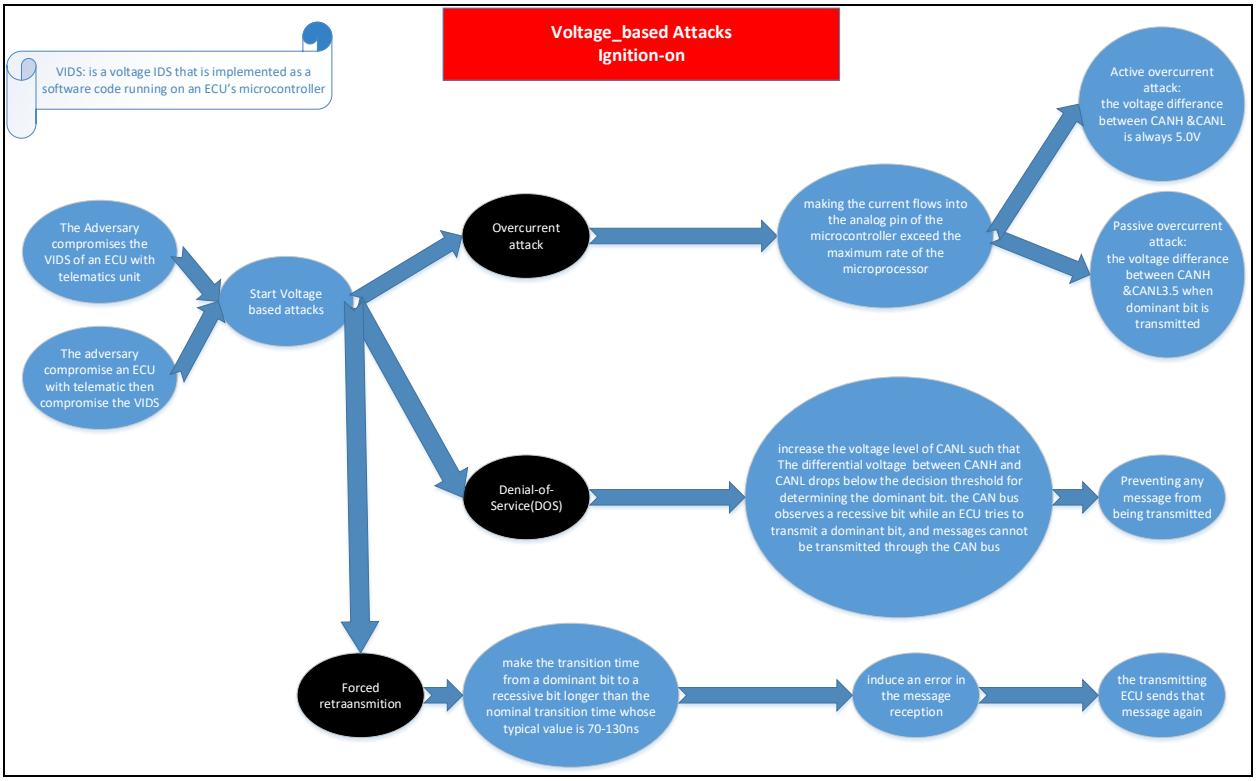

Figure 2: Voltage-based attacks

Message-based attacks

Masquerade attack [5] [6]

The attacker controls two malicious ECUs. The first one spoof the victim messages and the other one implement denial of service attack on the victim. This will lead the victim to Bus-off state and the

first malicious ECU will act the same like the victim. In this case, the malicious behavior will not be detected.

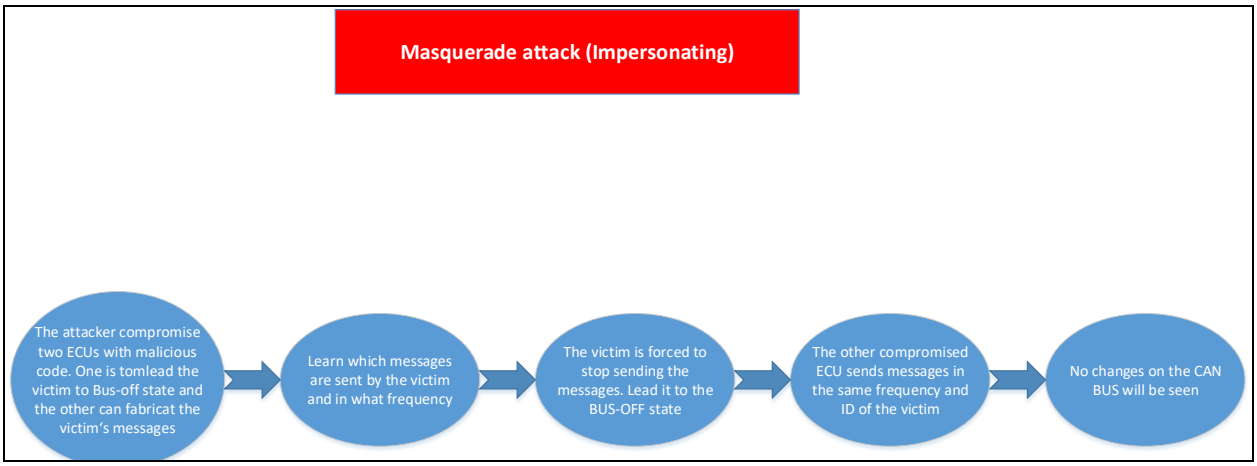

Figure 3: Masquerade attack

Injection [7]

1- Bad injection [8]

In a Bad Injection attack, the attacker injects a sequence of messages that never appeared before over the same CAN bus.

2- Replay attack [8] [9] [10] [11]

A Replay happens when an attacker injects in the CAN bus a sequence of messages that have previously been read from the same CAN bus

3- mixed injection [8]

Mixed Injection attacks are generated by injecting sequences that comprise a random mix of CAN messages

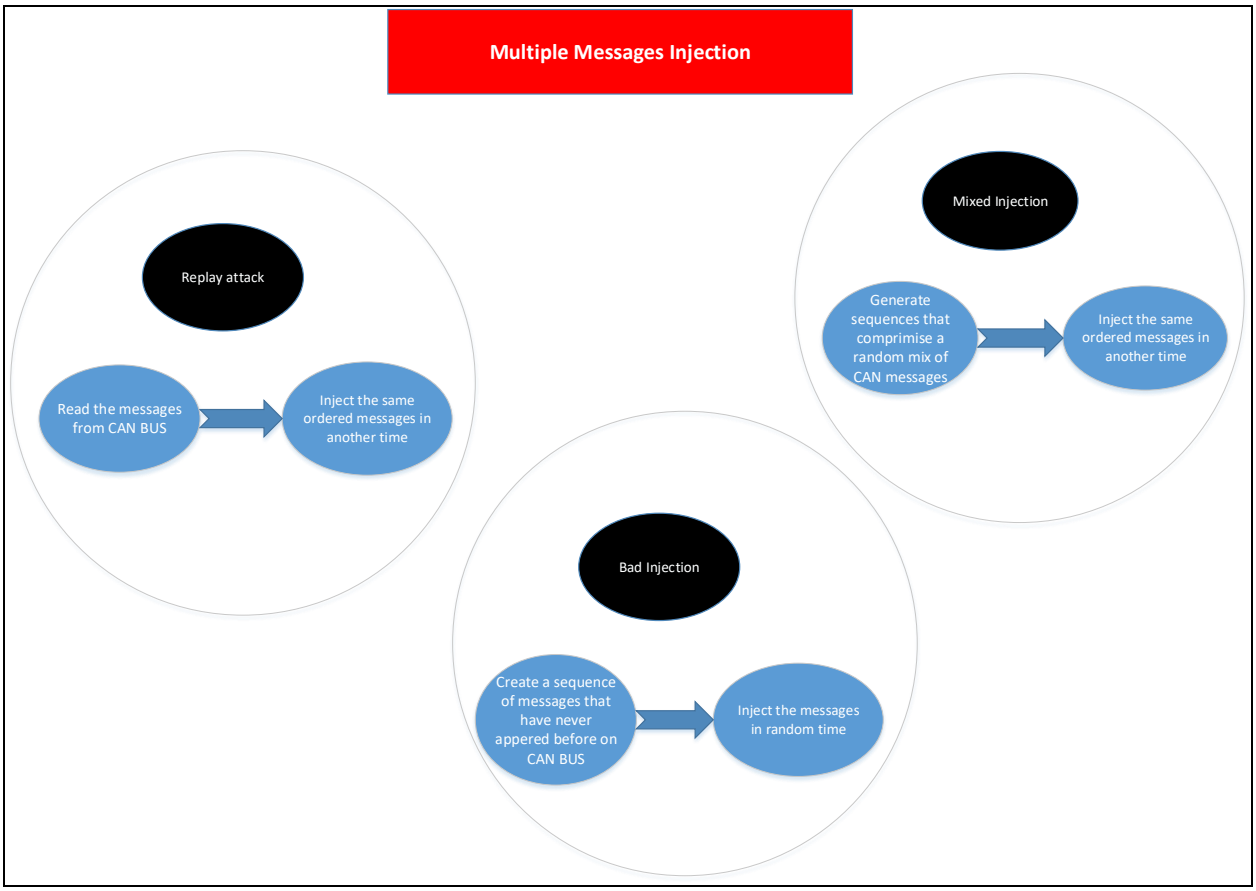

Figure 4: Multiple message injection

4- Fabrication attack [7] [5] [9] [10] [11] [8] [12]
Send messages on CAN bus with spoofed messages to change the behavior of specific receiving ECU.

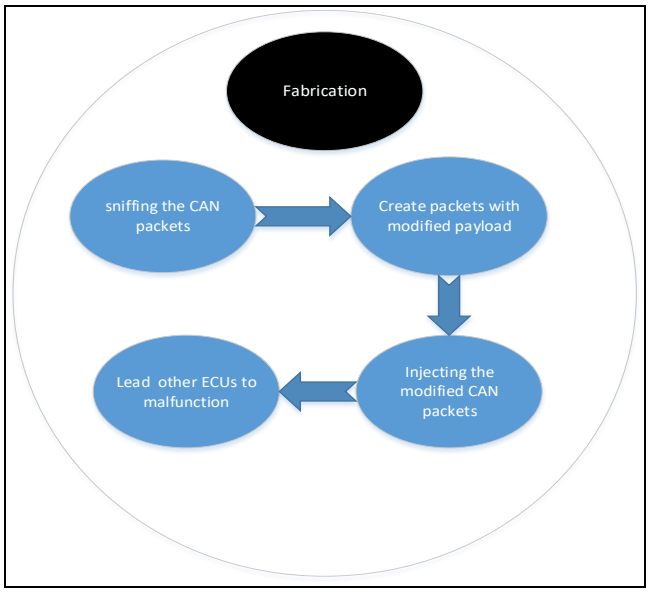

Figure 5: Fabrication

5- Fabrication attack by an attacker with voltage-timing Knowledge [13]

The adversary logs CAN traffic, learns the timing behavior of other ECUs, and exploits the learned information in injecting malicious messages at the appropriate (learned)times so as to imitate other ECUs' timing behavior. moreover the adversary should be capable of changing his voltage outputs via running battery draining processes, changing the supply voltage level, or by heating up or cooling down the ECU to impersonate another ECU.

Prerequisite: knowledge of how ECUs can be fingerprinted via voltage and timing measurements.

6- Fuzzing [14] [15] [16]

The strategy is, 1) sniffing the CAN packets with a freeware, 2) identifying CAN IDs used in the sniffed packets, 3) generating packets with CAN IDs contained in the sniffed packets, 4) fuzzing the data fields, 5) injecting the fuzzed CAN packets, and 6) monitoring the behavior of automobile.

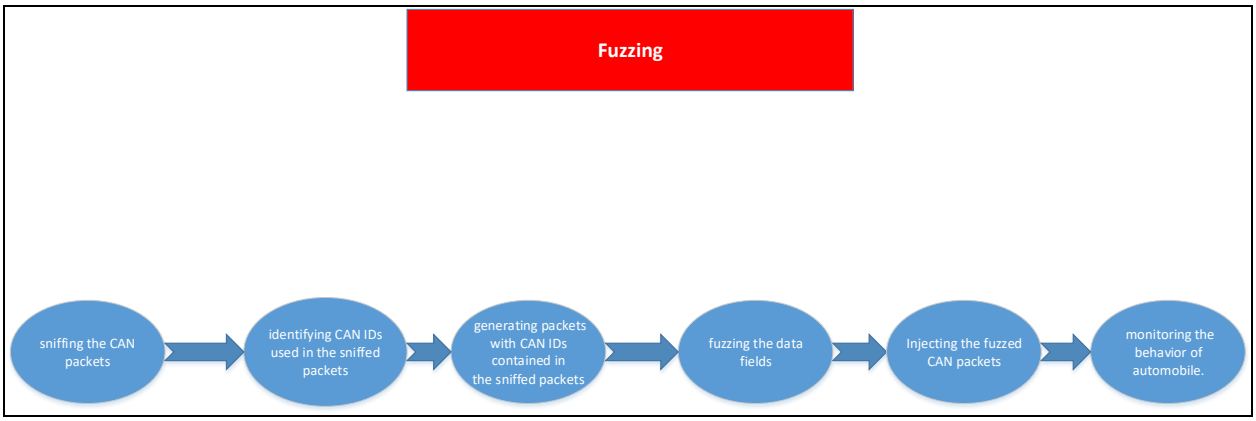

Figure 6: Fuzzing

7- Denial Of Service(DOS)

- ECU DOS [6]
  the attacker installs a malware in the Brake ECU (BECU) or floods it with wrong frames to put it out of service.
- BUS-DOS [6] [9] [11] [17]
  The attacker sends high priority messages on the bus massively to stop all the communication between ECUs.

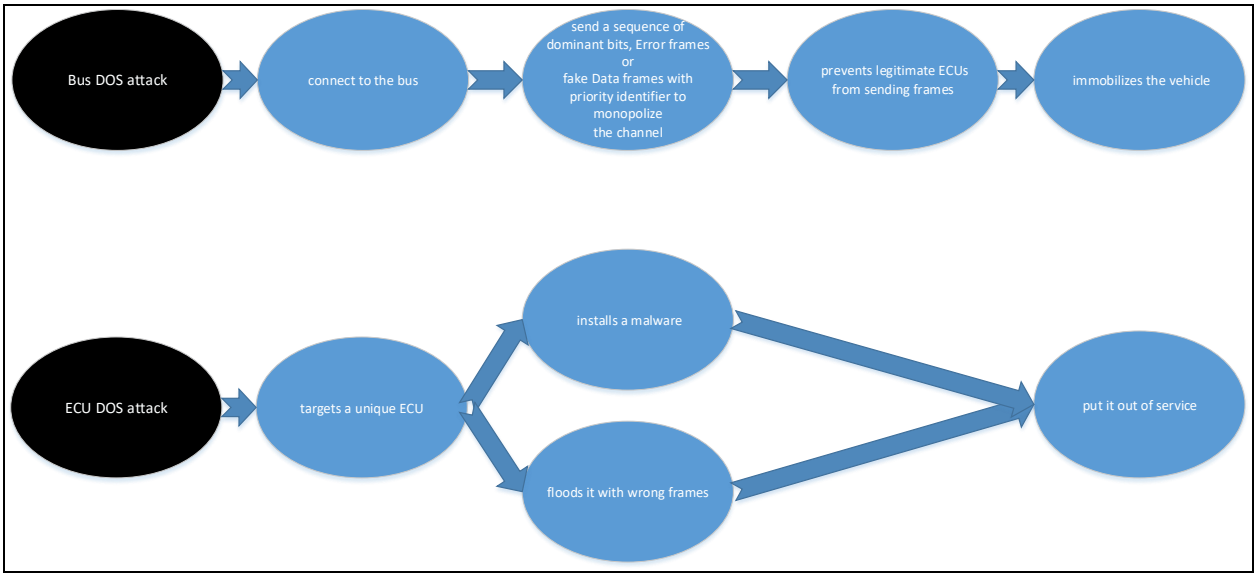

Figure 7: DOS attacks

8- Bus-Off attacks [18]
These attacks do not require sending entire frames onto the bus. An attacker can abuse the error handling mechanism[3] of the CAN protocol by sending few bits at the same time

[3] https://www.kvaser.com/about-can/the-can-protocol/can-error-handling/

the targeted ECU is transmitting a frame onto the bus. Consequently the corruption of the message will trigger the fault confinement mechanism[4] of CAN. After a certain number of errors, the targeted ECU will be put in "bus off" mode, preventing it from sending new frames. Bus-off attacks are possible:

- Using Bit error

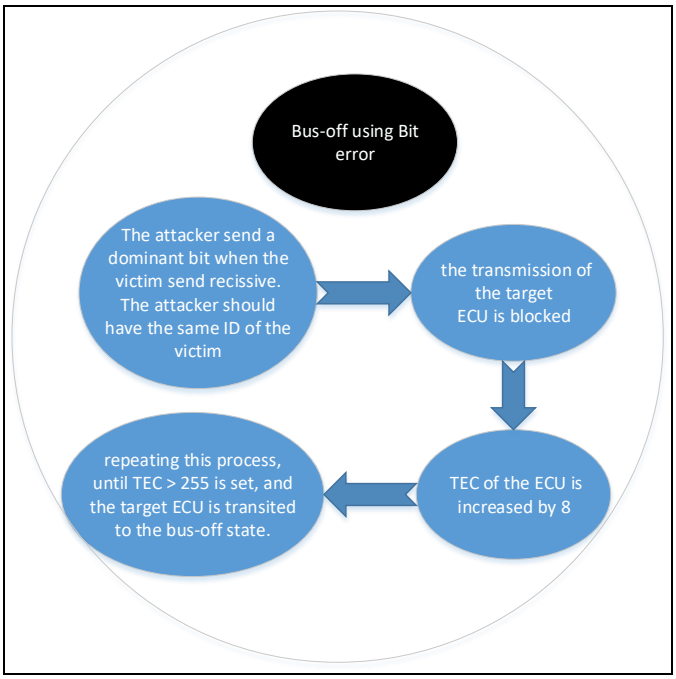

Figure 8: Bus-off using Bit error

- Using Stuff error

[4] https://embedclogic.com/can-protocol-protocol-to-broadcast-message-on-a-network/can-error-detection-and-fault-confinement/

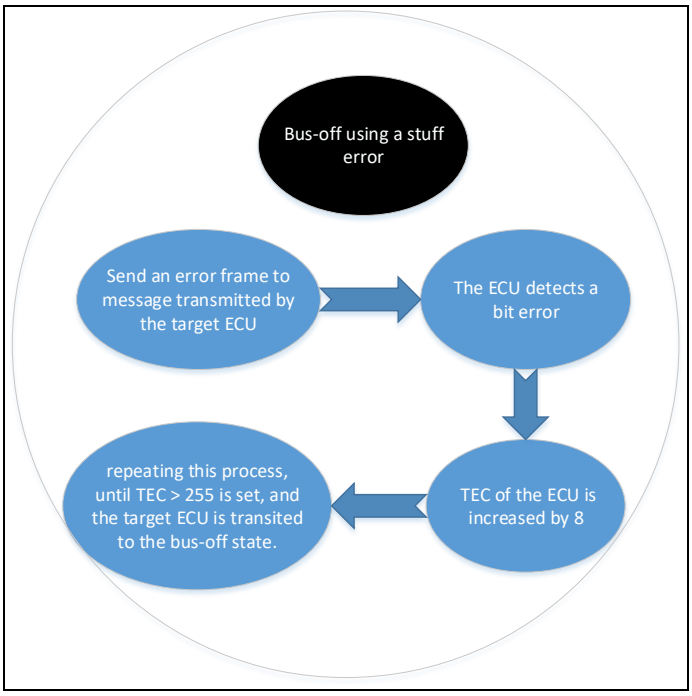

Figure 9: Bus-off using stuff error

- Using a frame

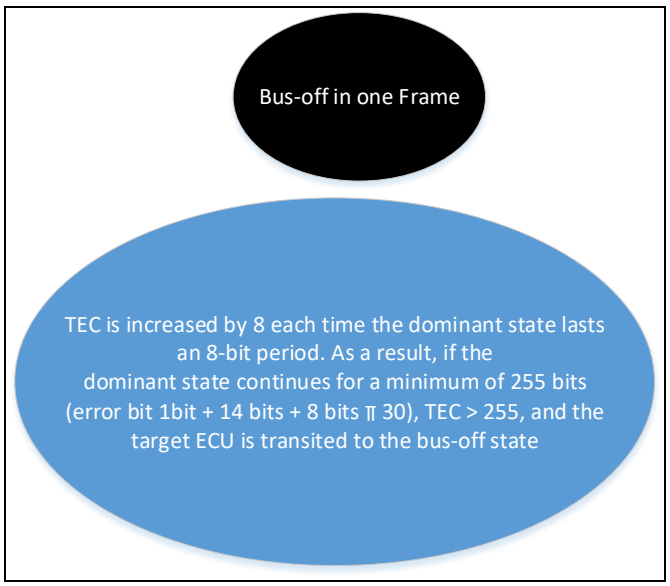

Figure 10: Bus-off using a frame

Manipulation

- Compromised Gateway attacks:
  - Steal messages [19]
  - Drop messages [20]
  - Modify messages [20]

Read Messages & Reverse engineering [14]

understand how certain hardware features are controlled by monitoring the changes when a specific message received by a specific ECU.

Malicious-code based attacks

Bridging Internal CAN [14]

Reprogram telematics unit which is connected to both low-speed and high-speed bus. Which make it possible to send packets from the low speed bus to high speed bus.

suspension [5]

To mount a suspension attack, the adversary needs only one weakly compromised ECU, i.e., become a weak attacker. As one type of Denial-of-Service (DoS) attack, the objective of this attack is to stop/suspend the weakly compromised ECU's message transmissions, so preventing the delivery/propagation of information it acquired, to other ECUs.

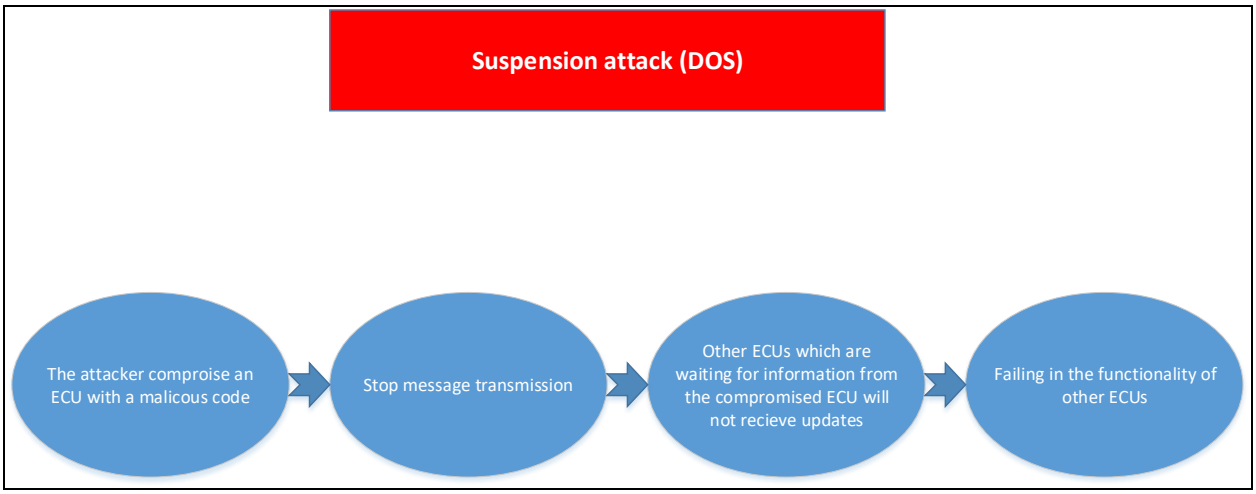

Figure 11: Suspension

Trojan

This kind of attacks trigger malicious code when there is a specific condition happens.

- Self-destruct [14]: "Self-Destruct" is a demo in which a 60-second count-down is displayed on the Driver Information Center (the dash), accompanied by clicks at an increasing rate and horn honks in the last few seconds. this sequence culminated with killing the engine and activating the door lock relay.
- Light-out(warning light) [14]: a Trojan runs when the car is driven in a specific speed. All the lights of the car turn off by sending fabricated messages on the CAN BUS.

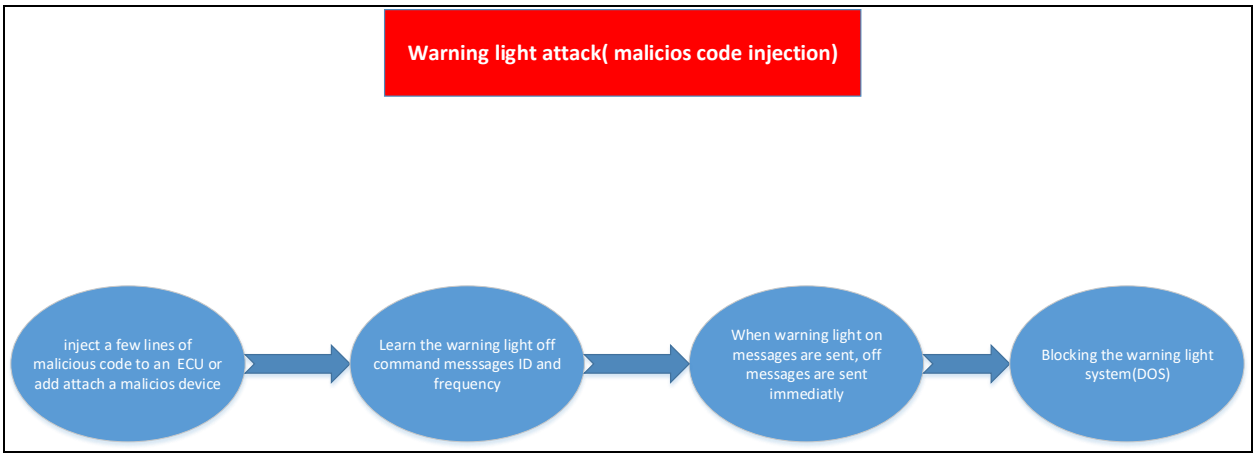

Figure 12: Warning light attack

- Lift-window [21]: a Trojan runs when the car is driven in a specific speed(200 kph). A lot of messages which make the lift window opens are sent on the Bus.

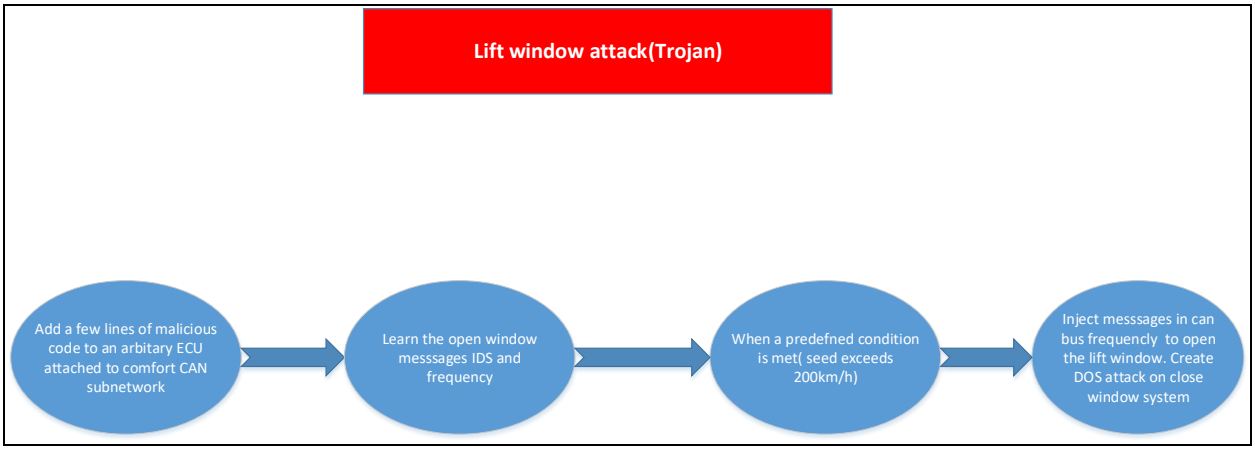

Figure 13: Lift-window attack

What are available defense mechanisms

There has been multiple solution proposed in literature and each of them has its own benefits and flaws. I can categorize them in four categories:

1- Cryptographic solutions [22](makes comparison between existent solutions)
2- Intrusion detection systems [23] [24] (makes comparison between existent solutions)
3- Intrusion reaction systems [25] [26] [27] [28] [29] [30] [31]

Parrot anti-spoofing defense system[31]	each ECU detects if its ID is used by another ECU	spoofing attack	send Dmessage(0x000000) faster than the spoofed message to lead the attacker to bus off state
Counter attack against the BUS-off attack[29]	number of error frames are seen on the bus in short time	Bus-off attack	perform bus off attack as a counter attack on the precceding frame which causes the bit error with the victim
Intrusion prevention system[28]	each ECU detects if its ID is used by another ECU and send alarm to the anomaly detector. The gateway listens to all messages and alert the detector if there is an invalid ID message	replay attack, invalid messages	send several dominant bit to make the unauthorized transmission message unvalid
Intrusion prevention system leveraging fault recovery[27]	use two algorithm to detect the attack before the messsage finishes transmission(Mesasge interval IDS,message response time analysis IDS)	messages injection	send error frame
using Centralized monitoring and interceptor ECU[26]	Pattern maching(CAN-ID,FIXED Payload) with anomaly detection(cycle, frequency, variable)	DOS, Spoofing attacks	send Error Frame to lead the attacker to bus-off state
A Method of prevention unauthorized data transmission[25]	each ECU detects if its ID is used by another ECU	Spoofing messages	send error frame before the unauhorized transmission is completed

Figure 14: Intrusion reaction systems

How to simulate CAN Bus and CAN attacks?

You can use Socket CAN to simulate an ECU, send messages on CAN bus and monitor the bus.
To simulate the whole networks and ECUs connected to it, you can use the powerful development and testing tool CANoe from Vector.

Summary

In this article we gained knowledge regarding CAN Bus protocol and possible attacks and what are the available solutions that can be implemented on CAN to secure the communication between ECUs.

Bibliography

[1] S. Hartzell and C. Stubel, "Automobile CAN Bus Network Security and Vulnerabilities," Seattle, Washington.

[2] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner and T. Kohno, "Comprehensive Experimental Analyses of Automotive Attack Surfaces," in *SEC'11: Proceedings of the 20th USENIX conference on Security*, San Diego, 2011.

[3] K.-T. Choy, Y. Kim and K. G. Shin, "Who Killed My Parked Car?," 23 January 2018. [Online]. Available: https://arxiv.org/abs/1801.07741.

[4] S. Uk Sagong, X. Ying, R. Poovendran and L. Bushnell, "Exploring Attack Surfaces of Voltage-Based Intrusion Detection Systems in Controller Area Networks," in *ESCAR'18*, Seattle, 2018.

[5] K.-T. Cho and K. G. Shin, "Fingerprinting Electronic Control Units for Vehicle Intrusion Detection," in *25th USENIX Security Symposium*, Austin, 2016.

[6] A. Boudguiga, W. Klaudel, A. Boulanger and P. Chiron, "A Simple Intrusion Detection Method for Controller Area Network," in *IEEE ICC 2016 Communication and Information Systems Security Symposium*, Kuala Lumpur, Malaysia, 2016.

[7] C. Valasek and C. Miller, "Adventures in Automotive Networks and Control Units," IOActive, 2014.

[8] M. Marchetti and D. Stabili, "Anomaly detection of CAN bus messages through analysis of ID sequences," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, Redondo Beach, CA, USA, IEEE, 2017.

[9] L. Pan, X. Zheng, H. Chen, T. Luan, H. Bootwala and L. Batten, "Cyber security attacks to modern vehicular systems," *Journal of Information Security and Applications,* 2017.

[10] Q. Wang and S. Sawhney, "VeCure: A Practical Security Framework to Protect the CAN Bus of Vehicles," in *2014 International Conference on the Internet of Things (IOT)*, Mountain View, USA, 2014.

[11] H. Min Song, H. R. Kim and H. K. Kim, "Intrusion Detection System Based on the Analysis of Time Intervals of CAN Messages for In-Vehicle Network," in *IEEE: 2016 International Conference on Information Networking (ICOIN)*, Seoul, Republic of Korea, 2016.

[12] D. K. VASISTHA, "DETECTING ANOMALIES IN CONTROLLER AREA NETWORK FOR AUTOMOBILES," Texas A&M University Libraries, TEXAS, 2017.

[13] K.-T. Cho and K. G. Shin, "Viden: Attacker Identification on In-Vehicle Networks," in *CCS '17: 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas Texas USA, Association for Computing Machinery, 2017, p. 1109–1123.

[14] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham and S. Savage, "Experimental Security Analysis of a Modern Automobile," in *2010 IEEE Symposium on Security and Privacy*, Oakland, California, USA, 2010.

[15] K.-T. Cho, K. G. Shin and T. Park, "CPS approach to checking norm operation of a brake-by-wire system," in *ICCPS '15: Proceedings of the ACM/IEEE Sixth*

International Conference on Cyber-Physical Systems, New York, NY, United States, Association for Computing Machinery, 2015, p. 41–50.

[16] H. Lee, K. Choi, K. Chung, J. Kim and K. Yim, "Fuzzing CAN Packets into Automobiles," in *2015 IEEE 29th International Conference on Advanced Information Networking and Applications (AINA)*, Gwangiu, South Korea, 2015.

[17] Q. Wang, Z. Lu and G. Qu, "An Entropy Analysis Based Intrusion Detection System for Controller Area Network in Vehicles," in *2018 31st IEEE International System-on-Chip Conference (SOCC)*, Arlington, VA, USA, 2018.

[18] K. Iehira, H. Inoue and K. Ishida, "Spoofing Attack Using Bus-off Attacks against a Specific ECU of the CAN Bus," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, USA, 2018.

[19] D. K. Oka, U. Larson and N. Larson, "Simulated Attacks on CAN Buses: Vehicle Virus," in *5th, IASTED Asian Conference on Communication Systems and Networks*, Langkawi, Malaysia, 2008.

[20] C.-W. Lin and A. Sangiovanni-Vincentelli, "Cyber-Security for the Controller Area Network (CAN) Communication Protocol," in *2012 International Conference on Cyber Security*, Washington, DC, 2012.

[21] H. Tobias, A. Lang, S. Kiltz and J. Dittmann, "Exemplary Automotive Attack Scenarios : Trojan Horses for Electronic Throttle Control System (ETC) and Replay Attacks on the Power Window System," 2007.

[22] M. Gmiden, M. Hedi Gmiden and H. Trabelsi, "Cryptographic and Intrusion Detection System for automotive CAN bus: Survey and contributions," in *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD'19)*, Istanbul, Turkey, 2019.

[23] S.-F. Lokman, A. T. Othman and M.-H. Abu-Bakar, "Intrusion detection system for automotive Controller Area Network (CAN) bus system: a review," 09 January 2018. [Online]. Available: https://doi.org/10.1186/s13638-019-1484-3. [Accessed 30 May 2019].

[24] N. SALMAN and M. BRESCH, Design and implementation of an intrusion detection system (IDS) for in-vehicle networks, Gothenburg, Sweden: Master thesis at Chalmers University of Technology, 2017.

[25] T. Matsumoto, M. Hata, M. Tanabe, K. Yoshioka and K. Oishi, "A Method of Preventing Unauthorized Data Transmission in Controller Area Network," in *2012 IEEE 75th Vehicular Technology Conference (VTC Spring)*, Yokohama, Japan , 2012.

[26] Y. Ujiie, T. Kishikawa, T. Haga, H. Matsushima, T. Wakabayashi, M. Tanabe, Y. Kitamura and J. Anzai, "A Method for Disabling Malicious CAN Messages by Using a CMI-ECU," SAE Technical Paper in United State, 2016.

[27] H. Olufowobi, S. Hounsinou and G. Bloom, "Controller Area Network Intrusion Prevention System Leveraging Fault Recovery," in *CPS-SPC'19: Proceedings of the ACM Workshop on Cyber-Physical Systems Security & Privacy*, New York, NY, United States, Association for Computing Machinery, 2019, p. 63–73.

[28] S. Abbott-McCune and L. Shay, "Intrusion prevention system of automotive network CAN bus," in *2016 IEEE International Carnahan Conference on Security Technology (ICCST)*, Orlando, FL, 2016.

[29] M. Takada, Y. Osada and M. Morii, " Counter Attack Against the Bus-Off Attack on CAN," in *2019 14th Asia Joint Conference on Information Security (AsiaJCIS)*, Kobe, Japan, 2019.

[30] T. Hoppe, S. Kiltz and J. Dittmann, "Adaptive Dynamic Reaction to Automotive IT Security Incidents Using Multimedia Car Environment," in *2008 The Fourth International Conference on Information Assurance and Security*, Naples, 2008.

[31] T. Dagan and A. Woo, "Parrot , a software-only anti-spoofing defense system for the CAN bus," 2016.