Demo: Attacks on CAN Error Handling Mechanism

Khaled Serag*, Vireshwar Kumar†, Z. Berkay Celik*, Rohit Bhatia*, Mathias Payer‡ and Dongyan Xu*
*Purdue University, {bhatia13, kserag, zcelik, dxu}@purdue.edu
†Indian Institute of Technology Delhi, viresh@cse.iitd.ac.in
‡EPFL, mathias.payer@nebelwelt.net

Attack Demo: https://youtu.be/aiNmhODx04U
Defense (RAID) Demo: https://youtu.be/CDHY1ZUwntU

***Abstract*—This demo shows how vulnerable CAN's error handling mechanism is by presenting three recent attacks that take advantage of this mechanism.**

We present three recent attacks that exploit CAN's error handling mechanism. The first attack is the *single frame bus off (SFBO)* attack, in which an attacker pushes a victim ECU to the bus off state by attacking a single frame. The bus off state is an error state in which an ECU disconnects itself from the bus. This attack exploits the way an ECU in the error passive state signals errors.

The second attack is the *persistent bus off* (*PBO*) attack, which uses and extends *SFBO*. This attack is facilitated by the fact that when the victim recovers, it sends the same message that caused it to enter the bus off state. The attacker picks a message and pushes its transmitter to the bus off state. The attacker then intercepts the victim's recovery and re-attacks it using the same technique, perpetually preventing its recovery.

The third attack is the *voltage corruption* attack *(*VC), in which an attacker injects fake messages while evading detection by *voltage intrusion detection systems (VIDS)*. This attack exploits a characteristic of the error passive state that allows an attacker to simultaneously transmit a message with another ECU without causing the two ECUs to abort transmission.

These attacks could be combined to allow an attacker to permanently disconnect an ECU from the bus while simultaneously impersonating its messages and evading detection by a (VIDS). We demonstrate how a *PBO* takes place by first running it on a testbed. Next, we evaluate all three attacks combined into one on a 2011 test-vehicle.

Testbed. We evaluate *SFBO* and *PBO* on a testbed composed of two Arduino Uno nodes, connected to the same CAN bus, operating at $500kbps$ through CAN bus shields. One plays the role of a victim transmitting two periodic IDs, and the other plays the role of the attacker. To monitor the bus traffic, we also attach a USB2CAN device. The attacker picks one of the two periodic message IDs and attacks it using *SFBO*, pushing the victim to the bus off state. The attacker then intercepts and re-attacks the victim's recovery using the same technique. This way, we turn a single instance of *SFBO* into a *PBO*. We achieve a 100% suppression rate of the victim.

Test Vehicle. We evaluate all three attacks, as shown in Fig. 1, on the high-speed CAN bus ($500kbps$) of a 2011 test vehicle. The bus contains 4 ECUs transmitting 50 periodic message IDs. Using the OBD port, we connect two Arduino Uno boards attached to CAN bus shields to act as the attacker and the accomplice. We attach an oscilloscope connected to a laptop acting as *Scission* [3] and we connect an additional ECU acting as *Viden* [4], two benchmark *VIDSs*. We pick the rpm message as our target. We first corrupt the voltage fingerprint during the training phase of the two VIDS. Next, we push its transmitter to the bus off state using *PBO* to completely prevent legitimate messages from appearing. Lastly, we inject fake rpm messages while evading detection by the VIDS. We achieve success rates reaching 95% against *Viden*, and 75% against *Scission*.

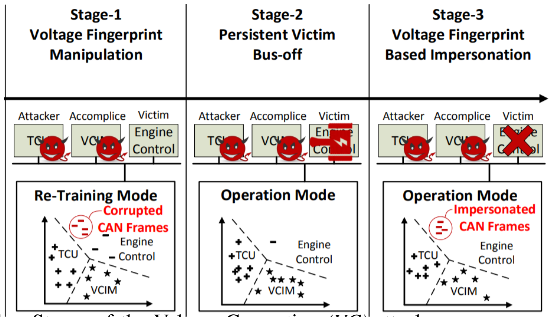

Fig. 1. Stages of the Voltage Corruption *(VC)* attack.

The root cause of all these attacks is a technique called *simultaneous transmission*, in which an attacker injects a message with the same ID as another message, at the exact same time. This causes an error, which the attacker uses to push the victim into different error states. To protect against these attacks, we propose a defense method called *RAndomized Identifier Defense (RAID)*. *RAID* turns standard IDs into an extended ones, then randomizes the padding. This way, if an attacker attempts to launch a *simultaneous transmission*, it fails since it does not know what ID to use. Since this approach directly tackles *simultaneous transmission*, it hence prevents the vast majority of CAN error handling attacks, and not only the ones proposed here. In the original papers [1], [2], we discuss the attacks as well as our defense *RAID* in full detail.

Acknowledgment

This work was supported in part by the Office of Naval Research (ONR) under Grant N00014-18-1-2674. Any opinions, findings, and conclusions in this paper are those of the authors and do not necessarily reflect the views of the ONR.

References

[1] Khaled Serag, Rohit Bhatia, Vireshwar Kumar, Z Berkay Celik, and Dongyan Xu. Exposing new vulnerabilities of error handling mechanism in CAN. In *USENIX Security Symposium*, 2021.

[2] Rohit Bhatia, Vireshwar Kumar, Khaled Serag, Z Berkay Celik, Mathias Payer, and Dongyan Xu. Evading voltage-based intrusion detection on automotive CAN. In *NDSS*, 2021.

[3] Marcel Kneib and Christopher Huth. Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2018.

[4] Kyong-Tak Cho and Kang G Shin. Viden: Attacker identification on in-vehicle networks. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2017.

Workshop on Automotive and Autonomous Vehicle Security (AutoSec) 2022
24 April 2022, San Diego, CA, USA
ISBN 1-891562-75-4
https://dx.doi.org/10.14722/autosec.2022.23013
www.ndss-symposium.org