Demo: A Security Infrastructure for Vehicular Information Using SDN, Intrusion Detection, and a Defense Center in the Cloud

Philipp Meyer*, Timo Häckel*, Falk Langer†, Lukas Stahlbock†, Jochen Decker‡,
Sebastian A. Eckhardt†, Franz Korf*, Thomas C. Schmidt*, and Fabian Schüppel†
**Dept. Computer Science*, *Hamburg University of Applied Sciences*, Germany
{philipp.meyer, timo.haeckel, franz.korf, t.schmidt}@haw-hamburg.de
†*IAV GmbH*, Berlin, Germany
{falk.langer, lukas.stahlbock, sebastian.alexander.eckhardt, fabian.schueppel}@iav.de
‡*easycore GmbH*, Erlangen, Germany
jochen.decker@easycore.com

***Abstract*—Vehicular on-board communication is the basis for advanced driver assistance, autonomous driving, over-the-air updates, and many more. If unprotected, this infrastructure is vulnerable to manipulation and various attacks. As any networked system, future connected cars require robust protection, monitoring, and incidence management against cyber-attacks during their lifetime. We demonstrate an infrastructure that secures the in-vehicle communication system and enables the security management of an entire vehicle fleet. Our prototype—a real-world production car—uses an Ethernet backbone network. It implements protective measures using software-defined networking, anomaly detection technologies, and is connected to a cyber defense center in the cloud. We demonstrate how this combination can reliably detect and mitigate common attacks on the vehicle—including its legacy components.**

***Index Terms*—Automotive, Networking, Security, IDS, SDN, Network Anomaly Detection, Cloud Defense Center**

I. Introduction

Ethernet-based In-Vehicle Networks (IVNs) are the future of vehicle on-board communication. They are the basis for new technologies such as IoT services, Over-the-Air (OTA) updates, advanced driver assistance systems and even autonomous driving. Current IVNs are vulnerable to manipulation by third parties, which has been shown in cyber-attacks in the field [1]. There are a multitude of interfaces for accessing devices in the car [2]. Manipulation of the IVN and its Electronic Control Units (ECUs) can impair the safety of the vehicle and thus even harm passengers. New regulations and guidelines demand features such as monitoring and incident management/response for the entire lifecycle of future vehicles [3], [4]. Multi-sided measures are therefore required to secure the safety critical IVN infrastructure.

In this work, we demonstrate a prototype of an IVN that enables protection, monitoring, detection, incidence management, and countermeasures in a real production car. The implementation uses Software-Defined Networking (SDN), Anomaly Detection (AD), secure gateways, hypervisors, dynamic orchestration of applications and cloud services.

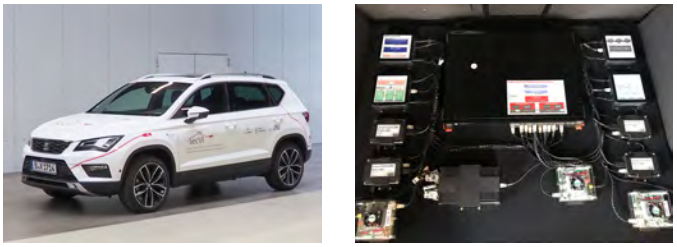
(a) 2016' Seat Ateca prototype (b) Installation in the trunk

Fig. 1: Pictures of the prototype and installed components

The following shows the installed components in our prototype vehicle (Section II), describes the showcases (Section III), and concludes this work (Section IV).

II. Prototype

Our prototype builds upon a Seat Ateca shown in Figure 1a. The ECUs of the car are grouped into five functional domains with one CAN bus per domain. Functions based on information from different domains require messages transferred from one domain to another via a central gateway.

We added an Ethernet zone-architecture by splitting the original domain CAN busses into four zones (front left, front right, rear left, rear right). In each zone, a Zone Controller (ZC) acts as a gateway between an Ethernet backbone and the CAN devices that are in its physical vicinity. We introduce additional native Ethernet communication such as high-resolution cameras sending raw camera images. Figure 1b shows our components installed in the trunk of the car and Figure 2 shows a conceptual drawing of our security infrastructure.

One exemplary ZC implements anomaly detection on CAN messages based on the timing specification of all CAN messages. For the separation of different domains on the zone controller itself, a hypervisor is used. Violations of the CAN message specification are reported via the Ethernet backbone.

This work is funded by the German Federal Ministry of Education and Research (BMBF) within the SecVI project.

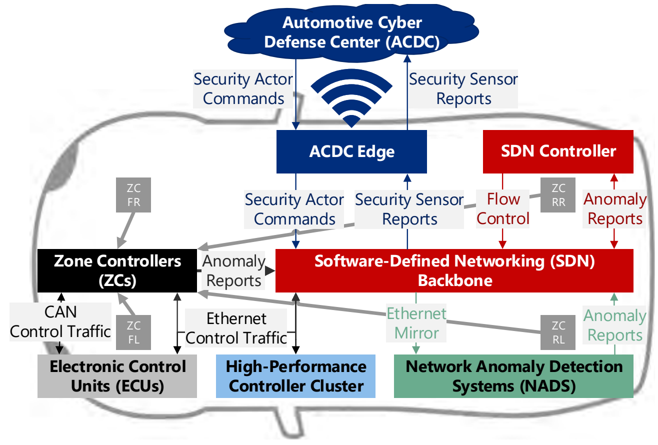

Fig. 2: Vehicle security infrastructure components

The Ethernet backbone consists of an OpenFlow-capable switch, which is divided into two virtual Open vSwitch instances (front, rear). An SDN controller controls the forwarding of frames in the flow tables of the network devices [5]. This enables precise access control on the network. Any mismatched frames are forwarded to the SDN controller, which can then decide to install a new flow rule or report the incident. Each Open vSwitch instance has a mirror port that mirrors all incoming traffic.

A Network Anomaly Detection System (NADS) monitors each mirror port. Each NADS uses machine learning to fingerprint the behavior of individual flows. Violations of the learned flow behavior are reported to the SDN controller and the cloud infrastructure.

Our Automotive Cyber Defense Center (ACDC) uses a cloud infrastructure to monitor the cybersecurity of large vehicle fleets and carry out incidence responses [6]. IoT Edge technologies allow the security management of each vehicle. By monitoring a vehicle fleet, attacks can be detected through the correlations of anomalies between multiple vehicles that cannot be detected by a single vehicle. The state of a vehicle's IT-infrastructure is monitored with security sensors in the car, e.g. CAN AD, NADS, and SDN. Security actuators enable the mitigation of attacks on vehicles, e.g. through network reconfiguration by the SDN controller or firewall reconfiguration.

Introduction of service-oriented architectures in automotive systems forms the basis for dynamic orchestration of applications as a security actuator mechanism. Three high-performance controllers connected to the Ethernet backbone are setup as a Lightweight Kubernetes (K3s) cluster. A master node allocates, and schedules containerized applications on computation nodes (slaves). In combination with an intrusion detection system the master monitors health and state of slaves and can change allocation of applications.

III. Showcases

Individual showcases illustrate the capabilities of the prototype and demonstrate aspects of the implemented mechanisms.

a) Manipulation on CAN bus: In the first showcase an attacker injects messages on a CAN bus connected to the ZC with anomaly detection. The ZC prevents the forwarding of the messages to other busses or the backbone and reports the incident to the ACDC.

b) Injection of unknown flows: This showcase demonstrates the protection of network infrastructure through software-defined separation of flows. A compromised network node tries to communicate via the backbone with flows that are unknown to the network. They are therefore not forwarded by the network devices and instead forwarded to the SDN controller. Incidents are reported to the ACDC.

c) Manipulation of known flows: In this scenario, the attacker knows which flows are forwarded via the software-defined backbone. A compromised node communicates with traffic that is defined in an established flow and is forwarded over the network. The NADSs connected to each virtual switch instance observe this specific flow and raise reports of detected anomalies. Each incident is forwarded to the SDN controller and the ACDC.

d) Incidence management in the cloud: Notifications from security sensors trigger observation alarms in the ACDC. If an incident is observed a defense center personal decides to put the reported vehicle into a secure mode. The enforcement of this secure mode within the vehicle is performed by the security actuators. For example, the SDN controller switches the network to a secure mode where only mandatory flows are permitted in the network.

e) Dynamic (re-)orchestration of applications: The last case shows a response on unavailable or compromised nodes. A node of the K3s cluster gets compromised. In response, two security actuator actions can change the allocation of applications on slaves. The first action reallocates applications from one slave to another. The second action stops and deletes a set of applications without reallocating them.

IV. Conclusion

Our demo showcases illustrate how a security infrastructure based on SDN, AD technologies, secure gateways, dynamic orchestration of applications, hypervisors and cloud services can enable protection, monitoring, detection, incidence management, and response for vehicles and entire fleets.

References

[1] C. Miller and C. Valasek, “Remote Exploitation of an Unaltered Passenger Vehicle,” *Black Hat USA*, vol. 2015, p. 91, 2015.

[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 *Proc. 20th USENIX Security Symp.*, vol. 4. USENIX Assoc., Aug. 2011, pp. 77–92.

[3] United Nations Economic Commission for Europe, “Draft Recommendation on Cyber Security of the Task Force on Cyber Security and Over-the-air issues of UNECE WP.29 GRVA,” UNECE, Standard WP.29, 2018.

[4] International Organization for Standardization, “Road vehicles – Cybersecurity engineering,” International Organization for Standardization, Geneva, CH, Standard ISO/SAE DIS 21434, 2020.

[5] T. Häckel, P. Meyer, F. Korf, and T. C. Schmidt, “Software-Defined Networks Supporting Time-Sensitive In-Vehicular Communication,” in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. Piscataway, NJ, USA: IEEE Press, Apr. 2019, pp. 1–5.

[6] F. Langer, F. Schüppel, and L. Stahlbock, “Establishing an Automotive Cyber Defense Center,” in *17th escar Europe : embedded security in cars*, 2019.