An Automotive Penetration Testing Framework for IT-Security Education

Stefan Schönhärl, Philipp Fuxen, Julian Graf, Jonas Schmidt, Rudolf Hackenberg and Jürgen Mottok
Ostbayerische Technische Hochschule, Regensburg, Germany
Email:{stefan1.schoenhaerl, philipp.fuxen}@st.oth-regensburg.de
Email:{julian.graf, jonas.schmidt, rudolf.hackenberg, juergen.mottok}@oth-regensburg.de

Abstract*—Automotive Original Equipment Manufacturer (OEM) and suppliers started shifting their focus towards the security of their connected electronic programmable products recently since cars used to be mainly mechanical products. However, this has changed due to the rising digitalization of vehicles. Security and functional safety have grown together and need to be addressed as a single issue, referred to as automotive security, in the following article. One way to accomplish security is automotive security education. The scientific contribution of this paper is to establish an Automotive Penetration Testing Education Platform (APTEP). It consists of three layers representing different attack points of a vehicle. The layers are the outer, inner, and core layers. Each of those contains multiple interfaces, such as Wireless Local Area Network (WLAN) or electric vehicle charging interfaces in the outer layer, message bus systems in the inner layer, and debug or diagnostic interfaces in the core layer. One implementation of APTEP is in a hardware case and as a virtual platform, referred to as the Automotive Network Security Case (ANSKo). The hardware case contains emulated control units and different communication protocols. The virtual platform uses Docker containers to provide a similar experience over the internet. Both offer two kinds of challenges. The first introduces users to a specific interface, while the second combines multiple interfaces, to a complex and realistic challenge. This concept is based on modern didactic theory, such as constructivism and problem-based learning. Computer Science students from the Ostbayerische Technische Hochschule (OTH) Regensburg experienced the challenges as part of a special topic course and provided positive feedback.

Keywords*—*IT-Security; Education; Automotive; Penetration testing; Education framework.

I. Introduction

Automotive security is becoming increasingly important. While OEM have developed vehicles for a long time with safety as a central viewpoint, security only in recent years started becoming more than an afterthought. This can be explained by bringing to mind, that historically vehicles used to be mainly mechanical products. With the rising digitalization of vehicles, however, the circumstances have changed.

Recent security vulnerabilities based on web or cloud computing services, such as Log4j, can be seen as entry points into vehicles, which an attacker can use to cause significant harm to the vehicle or people. To combat this, the development and release of new standards are necessary. The International Organization for Standardization (ISO) 21434 standard and United Nations Economic Commission for Europe (UNECE) WP.29, show the importance of automotive security in recent years.

However, there are other ways in which automotive security can be improved. Jean-Claude Laprie defines means of attaining dependability and security in a computer system, one of these being fault prevention, which means to prevent the occurrence or introduction of faults [1]. This can be accomplished by educating current and future automotive software developers. Since vulnerabilities are often not caused by systemic issues, but rather programmers making mistakes, teaching them about common vulnerabilities and attack vectors, security can be improved. Former research shows furthermore that hands-on learning not only improves the learning experience of participants but also increases their knowledge lastingly. Therefore, a framework for IT-security education has been developed, which was derived from penetration tests on modern vehicles.

The ANSKo was developed as an implementation of this framework. It is a hardware case, in which communicating Electronic Control Unit (ECU)s are simulated, while their software contains deliberately placed vulnerabilities. In a first step, users are introduced to each vulnerability, before being tasked with exploiting them themselves.

This paper aims at establishing a realistic and effective learning platform for automotive security education. Therefore, the following research questions are answered:

- (RQ1) - What content is appropriate for an automotive penetration testing framework for IT-security education?
- (RQ2) - How could an automotive security education platform be implemented?

The structure of the paper starts with the related work in Section II. Section III introduces an architecture derived from modern vehicle technologies. Those technologies are then classified into layers and briefly explained in Section IV. The structure and used software of the ANSKo itself are presented in Section V. Section VI presents the learning concept and its roots in education theory. The paper ends with a conclusion in Section VII.

II. Related Work

Hack The Box (HTB) is a hands-on learning platform with several vulnerable virtual systems that can be attacked by the user. Thereby, a big focus of this platform is gamification.

TABLE I
COMPARISON OF THE DIFFERENT APPROACHES

	HTB	HaHa SEP	RAMN	ANSKo
Virtual approach	YES	NO	NO	YES
Hardware approach	NO	YES	YES	YES
Automotive specific	NO	NO	YES	YES
Gamification	YES	NO	NO	YES
IT-Security	YES	YES	YES	YES

They do not offer automotive-specific systems and access to physical hardware is also not possible [2]. One approach that focuses on hardware-specific attacks is the Hardware Hacking Security Education Platform (HaHa SEP). It provides practical exploitation of a variety of hardware-based attacks on computer systems. The focus of HaHa SEP is on hardware security rather than automotive security. Students who are not present in the classroom can participate via an online course. A virtual version of the hardware cannot be used [3]. The Resistant Automotive Miniature Network (RAMN) includes automotive and hardware-related functions. The hardware is very abstract and is located on a credit card-sized Printed Circuit Board (PCB). It provides closed-loop simulation with the CARLA simulator but there is no way to use RAMN virtually. The focus of RAMN is to provide a testbed that can be used for education or research. However, it is not a pure education platform [4].

The fundamental and related work for the APTEP are real-world attack patterns. The technologies used for connected vehicles represent a particularly serious entry point into the vehicle, as no physical access is required. Once the attacker has gained access to the vehicle, he will attempt to penetrate further into the vehicle network until he reaches his goal. This can be done with a variety of goals in mind, such as stealing data, stealing the vehicle, or even taking control of the vehicle. The path along which the attacker moves is called the attack path. Such a path could be demonstrated, for example, in the paper ”Free-Fall: Hacking Tesla from wireless to Controller Area Network (CAN) Bus” by Keen Security Labs. The researchers succeeded in sending messages wirelessly to the vehicle’s CAN bus [5]. The same lab was also able to show further vulnerabilities, e.g., Bluetooth, Global System for Mobile Communications (GSM), and vehicle-specific services [6]. Valasek and Miller demonstrated the vulnerability of a vehicle’s infotainment system [7]. Using various attack paths, they managed to make significant changes to the vehicle.

Teaching at universities is often theory-based. As a result, many graduates may lack the practical experience to identify vulnerabilities. But it is precisely this experience that is of great importance in the professional field of software development, security testing, and engineering. The idea is to develop the competence level from a novice to an experts level, which can be guided by ”Security Tester” certified Tester Advanced Level Syllabus. The described APTEP presents an ecosystem to establish such learning arrangements in which constructivism-based learning will happen [8][9].

III. ARCHITECTURE

The attacks from the previous section show, that attacks follow a similar pattern. There is an entry point through which the attacker gains access to the vehicle. He then tries to move through the vehicle network by exploiting further vulnerabilities. He does this until he reaches his target. To represent this procedure in the architecture of ANSKo, it was divided into different layers.

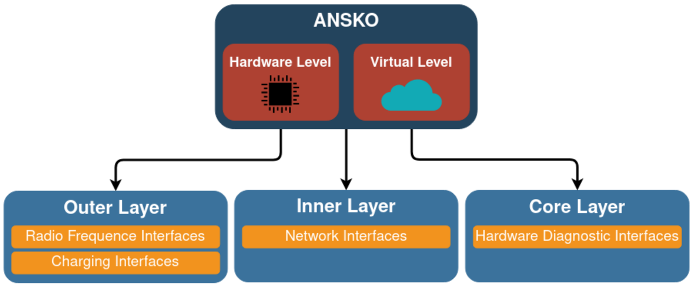

Fig. 1. ANSKO Architecture

As shown in Figure 1, the following three layers were chosen: Outer layer, inner layer, and core layer. They delimit the respective contained interfaces from each other.

A. Outer Layer

The automotive industry is currently focusing heavily on topics, such as automated driving functions, Vehicle-to-Everything (V2X) networking, and Zero-Emission Vehicles (ZEV). In these areas, new trend technologies can lead to valuable new creations. But unfortunately, this development also favors the emergence of new and more critical points of attack. For this reason, the outer layer was included in the APTEP as part of the architecture. It contains all the functionalities that enable the vehicle to communicate with its environment. This includes the two V2X technologies Cellular-V2X and WLAN-V2X as well as other communication protocols, such as Bluetooth and GSM. In addition to the communication protocols, there are also interfaces, such as various charging interfaces, sensors, and much more.

The outer layer represents an important component because many interfaces contained in it represent a popular entry point for attacks. This is the case because the technologies used there are usually an option to potentially gain access to the vehicle without having physical access to the vehicle. Even if the sole exploitation of a vulnerability within the outer layer does not always lead to direct damage in practice, further attack paths can be found over it. In most cases, several vulnerabilities in different areas of the vehicle system are combined to create a critical damage scenario from the threat. Therefore, vehicle developers need to be particularly well trained in this area.

B. Inner Layer

The inner layer of the APTEP represents the communication between individual components. While modern vehicles implement different forms of communication, bus systems like Controller Area Network (CAN), Local Interconnect Network (LIN), and FlexRay used to be predominant. Since modern vehicle functions connected to the Outer Layer, like image processing for rearview cameras or emergency braking assistants [10], require data rates not achievable by the previously mentioned bus systems, new communication systems, like Ethernet, have been implemented in vehicles.

Depending on the scope, the mentioned bus systems are still in use because of their low cost and real-time capabilities. From those communication technologies, different network topologies can be assembled. Individual subsystems connecting smaller components, e.g., ECUs, are themselves connected through a so-called backbone. Gateways are implemented to connect the subsystems with the backbone securely.

After gaining access to a vehicle through other means, the inner layer represents an important target for attackers since it can be used to manipulate and control other connected components. While the target components can be part of the same subsystem, it is also possible, that it is part of a different subsystem, forcing the attacker to communicate over the backbone and the connected gateways. The inner layer thus represents the interface between the outer - and core layer.

C. Core Layer

Manipulating the ECUs of a vehicle themselves results in the greatest potential damage and therefore represents the best target for a hacker. In the APTEP, this is represented as the core layer.

Vehicles utilize ECUs in different ways, e.g., as a Body Control Module, Climate Control Module, Engine Control Module, Infotainment Control Unit, Telematic Control Unit. In addition, electric vehicles include further ECUs for special tasks, such as charging.

If attacks on an ECU are possible, its function can be manipulated directly. Debugging and diagnostic interfaces, like Joint Test Action Group (JTAG) or UDS (Unified Diagnostic Services), are especially crucial targets since they provide functions for modifying data in memory and reprogramming of ECU firmware.

The impact of arbitrary code execution on an ECU is dependent on that ECUs function. While taking over, e.g., a car's infotainment ECU should only have a minor impact on passengers' safety, it can be used to attack further connected devices, via inner layer, from an authenticated source. The goal of such attack chains is to access ECUs where safety-critical damage can be caused. Especially internal ECUs interacting with the engine can cause severe damage, like shutting off the engine or causing the vehicle to accelerate involuntarily.

IV. Interfaces

This section describes some chosen interfaces of the previously presented layers. The selection was made from the following three categories: ”Radio Frequency and Charging Interfaces”, ”Network Interfaces” and ”Hardware Diagnostic Interfaces”. Implemented in the ANSKo is one interface from each architecture layer - NFC from the outer layer (Section IV-A1c, CAN from the inner layer (Section IV-B1), and UDS from the core layer (Section IV-C2). This facilitates the cross-domain challenges described in Section VI.

A. Radio Frequency and Charging Interfaces

The outer layer contains the interfaces of the category ”radio frequency and charging interfaces”. They all have in common that they enable the vehicle to communicate with its environment. Furthermore, the included interfaces can be divided into the following classes: short-range communication, long-range communication, and charging interfaces.

1) Short-range Communication:

a) Bluetooth: Bluetooth is a radio standard that was developed to transmit data over short distance wireless. In the vehicle, the radio standard is used primarily in the multimedia area. A well-known application would be, for example, the connection of the smartphone to play music on the vehicle's internal music system.

b) RFID: Radio frequency identification (RFID) enables the communication between an unpowered tag and a powered reader. A powered tag makes it possible to increase the readout distance. RFID is used, for example, in-vehicle keys to enable keyless access.

c) NFC: Near field communication (NFC) is an international transmission standard based on RFID. The card emulation mode is different from RFID. It enables the reader to also function as a tag. In peer-to-peer mode, data transfer between two NFC devices is also possible. In vehicles, NFC is used in digital key solutions.

d) WLAN-V2X: The WLAN-V2X technology is based on the classic WLAN 802.11 standard, which is to be used in short-range communication for V2X applications. However, almost all car manufacturers tend to focus on Cellular-V2X because long-range communication is also possible in addition to short-range communication.

2) Long-range Communication:

a) GNSS: The Global Navigation Satellite System (GNSS) comprises various satellite navigation systems, such as the Global Positioning System (GPS), Galileo, or Beidou. Their satellites communicate an exact position and time using radio codes. In vehicles, GNSS is mainly used in onboard navigation systems. Furthermore, it is increasingly used to manage country-specific services.

b) Cellular-V2X: Cellular-V2X forms the communication basis for V2X applications. It uses the cellular network for this purpose. In contrast to WLAN-V2X, it enables both V2V and vehicle-to-network (V2N) communication.

3) Charging Interfaces: To enable charging or communication between an electric vehicle and a charging station, a charging interface is required. Due to the high diversity in this area, there is not just one standard.

a) CHAdeMO: The CHAdeMO charging interface was developed in Japan where it is also used. The charging process can be carried out with direct current (DC) charging. Mainly

Japanese OEMs install this charging standard in their vehicles. Some other manufacturers offer retrofit solutions or adapters.

b) Tesla: Tesla predominantly uses their own charging interface, which allows both alternating current (AC) and DC charging. However, due to the 2014/94 EU standard, Tesla is switching to the Combined Charging System (CCS) Type-2 connector face in Europe.

c) CCS: The official European charging interfaces CSS Type-1 and CSS Type-2 are based on the AC Type-1 and Type-2 connectors. The further development enables a high DC charging capacity in addition to the AC charging.

B. Network Interfaces

Network interfaces describe the technologies used to communicate between components, like ECUs or sensors. It represents the inner layer.

1) CAN: CAN is a low-cost bus system, that was developed in 1983 by Bosch. Today it is one of the most used bus systems in cars since it allows acceptable data rates of up to 1Mbit/s while still providing real-time capabilities because of its message prioritization. Its design as a two-wire system also makes it resistant to electromagnetic interference.

Traditionally in a vehicle CAN is often used as the backbone, providing a connection between the different subsystems. It is also used in different subsystems itself, like engine control and transmission electronics.

2) LIN: The LIN protocol was developed as a cost-effective alternative to the CAN bus. It is composed of multiple slave nodes, which are controlled by one master node, which results in a data rate of up to 20Kbit/s.

The comparatively low data rate and little fault resistance result that LIN being mainly used in non-critical systems, like power seat adjustment, windshield wipers, and mirror adjustment.

3) MOST: The Media Oriented System Transport (MOST) bus provides high data rates of 25, 50, or 150 Mbit/s depending on the used standard. It was developed specifically for use in vehicles and is typically implemented as a ring.

As the name suggests the field of application for the MOST bus is not in safety-critical systems. but in multimedia systems of a vehicle. Since transmission of uncompressed audio and video data requires high data rates, MOST are suited best for those tasks.

4) FlexRay: FlexRay offers data transmission over two channels with 10Mbit/s each. They can be used independently or by transmitting redundant data for fault tolerance. Furthermore, FlexRay implements real-time capabilities for safety-critical systems.

FlexRay was developed with future X-by-Wire (steer, brake, et al.) technologies in mind [11]. Even though FlexRay and CAN share large parts of their requirements, FlexRay improves upon many aspects, leading to it being used as a backbone, in powertrain and chassis ECUs and other safety-critical subsystems.

5) Ethernet: Automotive Ethernet provides a cost-effective transmission protocol with high data rates of 1Gbit/s. While the underlying Ethernet protocol is not fit to be used in systems with electromagnetic interference and also offers no real-time capabilities, this can be remedied by using the BroadR-Reach and Audio-Video-Bridging (AVB) standards respectively.

Due to the constant increase in required data rates in new technologies, such as image processing, Ethernet was adapted for its use in vehicles. Because of its widespread use even outside of vehicles, it offers many different protocols, which are constantly being improved.

C. Hardware-Diagnostic Interfaces

The hardware-diagnostic interfaces are classified in the core layer. They describe technologies, that allow interaction between a person, such as a programmer, and an ECU to allow, e.g., reprogramming of the software.

1) Debug: Debug interfaces are used in embedded development to allow debugging, reprogramming, and reading out error memory of the circuit boards. Vehicles implement various debug interfaces, depending on their integrated circuit boards. The most common interfaces include Joint Test Action Group (JTAG), Serial Wire Debug (SWD), Universal Asynchronous Receiver Transmitter (UART), and Universal Serial Bus (USB).

Interacting with the debug interfaces requires special equipment, like adapters.

2) UDS: Modern vehicles implement a diagnostic port as well to allow independent car dealerships and workshops functionalities similar to the debug interfaces while not being unique to one particular OEM. It uses the communication protocol Unified Diagnostic Services (UDS), defined in the ISO 14229 standard.

UDS utilizes CAN as the underlying protocol to transmit messages. To prevent unauthorized access to the diagnostic port, UDS provides different tools, like ”Diagnostic Session Control” which defines different sessions, such as default, diagnostic, or programming. OEMs can choose which service is available in each session. Security-critical services can also be further guarded by using the ”Security Access” which protects the respective service through a key seed algorithm.

3) Side Channels: The final interface in the core layer are side channels. A computing unit emits certain side-channel data while performing operations, such as the consumed energy while encrypting data. They allow attackers to gain information about secret parts of the computer system like the used keys for cryptographic operations. Side-channel data can therefore be used to attack otherwise secure computer systems. Possible different side channels include time, power, fields, and temperature.

V. Structure

The presented APTEP is implemented in the ANSKo, which consists of a hardware and a virtual level. Their required components and used software are described in the following.

A. *Hardware-Level*

The goal of the ANSKo is to provide a low-cost learning environment for automotive security. A picture of the hardware contents can be seen in Figure 2. The currently included components are marked by colors. It is intended to further extend the platform by the listed interfaces in Section IV.

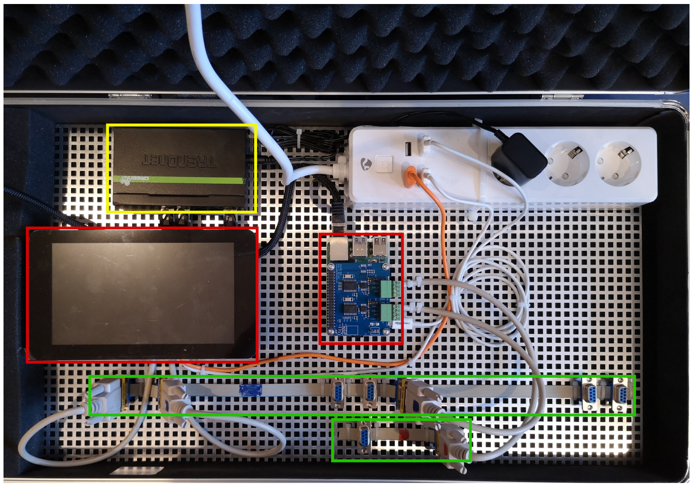

Fig. 2. ANSKo Hardware

- **Yellow - Ethernet Switch:** The Ethernet switch connects to both Raspberry Pis and allows additional connection to the user.
- **Red - Display and Raspberry Pis:** The main components of the case are two Raspberry Pis, which simulate ECUs in a vehicle. They possess a PiCAN Duo board allowing two independent CAN connections. One of the Raspberry Pis possesses a display, simulating a dashboard with a speedometer and other vehicle-specific values.
- **Green - CAN Bus:** The CAN Bus is the main communication channel in the current structure. Connected devices can be disconnected by removing the respective cables.

One implemented challenge in the ANSko is a Man-in-the-Middle attack. The goal is to lower the displayed mileage of the car to increase its value. A user working with the ANSKo needs to read the messages being sent between the simulated ECUs. They can interact with the CAN Bus by connecting to the CAN Bus via USB cable and an included Embedded60 microcontroller.

The operating system running on the Raspberry Pis was built by using pi-gen It allows generating and configuring a Raspberry Pi OS image. By using the automation software Ansible, challenges can be installed on all cases simultaneously. Challenges are started as a systemd service after copying the required files to the cases.

B. *Virtual-Level*

During the Covid-19 pandemic holding education courses hands-on was not possible. To still provide the advantages of the ANSKo during lockdowns, an online platform with identical challenges has been realized.

The virtual challenges are accessible through a website, which allows the authentication of users. A user can start a challenge, which creates a Docker container. This ensures an independent environment for users while also protecting the host system. Users can receive the necessary CAN messages by using the socketcand package, providing access to CAN interfaces via Transmission Control Protocol/Internet Protocol (TCP/IP).

The unique docker containers for each user allow them to stop and start working on the challenge at any time but limits the maximum amount of users attempting the challenges concurrently. Validation of a correct solution also does not have to be carried out manually by sending a unique string of characters on the CAN bus which can be compared to the back end by the user.

VI. Learning Concept

ANSKo's concept of learning is based on the theory of constructivism. It allows learners to achieve the higher-order learning goals of Bloom's Taxonomy. They are more capable of analyzing facts and problems, synthesizing known information, and evaluating their findings [12].

Learning concepts are used to encourage learners to actively think rather than passively absorb knowledge, e.g., Problem-Based Learning (PBL). ANSKo consists of several real-world problems, so-called challenges. Support for problem-solving uses the scaffolding approach, i.e., learners initially receive theoretical knowledge, optimize their learning progress in groups, and solve the problem independently [12].

The challenges can be divided into two categories: ”Domain-specific challenges” and ”Cross-domain challenges”. The two types each pursue different learning objectives.

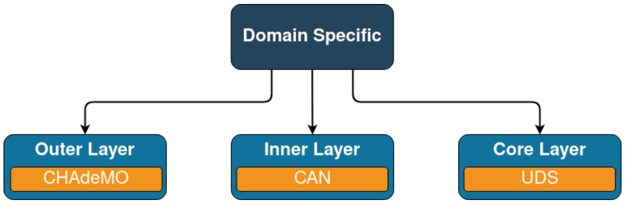

Fig. 3. Domain-specific Challenge

As shown in Figure 3, ”Domain-specific challenges” are about learning the functionalities and vulnerabilities of a single interface within a domain. A challenge is considered complete when the learner has found and exploited the vulnerability.

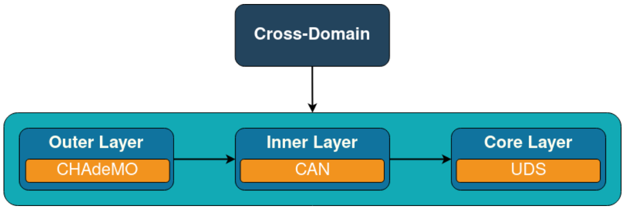

Fig. 4. Cross-domain Challenge

Cross-domain challenges aim to teach the learner how to find and exploit attack paths. Figure 4 shows an example of a cross-domain challenge. Here, interfaces from the different layers are combined. The difficulty level of these challenges is higher and therefore the respective domain-specific challenges for the required interfaces have to be solved first.

Computer science students from the OTH Regensburg were able to work with the ANSKo as part of a special topic course for the 6th & 7th semesters. The course evaluation, which was answered by the students, showed the benefit of the learning platform. They reported a positive experience when working with the ANSKo, e.g., when asked about understanding the importance of automotive security or their learning progress. The selected challenges were quoted as adequately difficult to be solved using the underlying learning concept.

VII. Conclusion

The presented vulnerabilities at the beginning of this paper and the listing of strengths and weaknesses of existing learning platforms justify the need for an automotive-specific IT security learning platform. For this reason, an APTEP was developed on which participants can learn about vulnerabilities in practice.

To realize this, an architecture for the APTEP was chosen that maps the described attacks. The architecture consists of three layers - outer layer, inner layer, and core layer. Each of them contains different interfaces, such as the Radio Frequency interface as well as the Charging interface in the outer layer, Network interfaces in the inner layer, and Hardware-Diagnostic interfaces in the core layer.

The APTEP is implemented on the Hardware level to provide a realistic learning environment, but also offers a virtual level, which allows users to work with the platform remotely since the Covid-19 pandemic prevented hands-on work.

To keep the challenges as realistic as possible while providing learners with an appropriate level of complexity, the tasks were divided into two categories. There are ”Domain-specific challenges,” which deal with only one interface per challenge. A ”Cross-domain challenge” cannot be solved until the associated ”Domain-specific challenges” have been solved for each included interface. The ”Cross-domain challenges” combine different interfaces and teach learners to find and exploit attack paths.

Future work includes the implementation of electric vehicle-specific challenges, e.g., charging interfaces. Side-channel attack challenges will be included as well.

To support the individual learning progress eye tracking will be included and analyzed. The learner’s cognitive load will be determined by AI-based classification results. Finally, this will improve individual learning success.

References

[1] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr, “Basic concepts and taxonomy of dependable and secure computing,” 2004.

[2] H. the Box, *Hack the box*. [Online]. Available: https://www.hackthebox.com/ (retrieved: 02/2022).

[3] S. Yang, S. D. Paul, and S. Bhunia, “Hands-on learning of hardware and systems security.,” *Advances in Engineering Education*, vol. 9, no. 2, n2, 2021. [Online]. Available: https://files.eric.ed.gov/fulltext/EJ1309224.pdf (retrieved: 02/2022).

[4] C. Gay, T. Toyama, and H. Oguma, “Resistant automotive miniature network,” [Online]. Available: https://fahrplan.events.ccc.de/rc3/2020/Fahrplan/system/event_attachments/attachments/000/004/219/original/RAMN.pdf (retrieved: 02/2022).

[5] S. Nie, L. Liu, and Y. Du, “Free-fall: Hacking tesla from wireless to can bus,” *Briefing, Black Hat USA*, vol. 25, pp. 1–16, 2017. [Online]. Available: https://www.blackhat.com/docs/us-17/thursday/us-17-Nie-Free-Fall-Hacking-Tesla-From-Wireless-To-CAN-Bus-wp.pdf (retrieved: 02/2022).

[6] Z. Cai, A. Wang, W. Zhang, M. Gruffke, and H. Schweppe, “0-days & mitigations: Roadways to exploit and secure connected bmw cars,” *Black Hat USA*, vol. 2019, p. 39, 2019. [Online]. Available: https://i.blackhat.com/USA-19/Thursday/us-19-Cai-0-Days-And-Mitigations-Roadways-To-Exploit-And-Secure-Connected-BMW-Cars-wp.pdf (retrieved: 02/2022).

[7] C. Miller and C. Valasek, “Remote exploitation of an unaltered passenger vehicle,” *Black Hat USA*, vol. 2015, no. S 91, 2015.

[8] F. Simon, J. Grossmann, C. A. Graf, J. Mottok, and M. A. Schneider, *Basiswissen Sicherheitstests: Aus- und Weiterbildung zum ISTQB® Advanced Level Specialist – Certified Security Tester*. dpunkt.verlag, 2019.

[9] International Software Testing Qualifications Board, *Certified tester advanced level syllabus security tester, international software testing qualifications board*, 2016. [Online]. Available: https://www.german-testing-board.info/wp-content/uploads/2020/12/ISTQB-CTAL-SEC_Syllabus_V2016_EN.pdf (retrieved: 02/2022).

[10] P. Hank, S. Müller, O. Vermesan, and J. Van Den Keybus, “Automotive ethernet: In-vehicle networking and smart mobility,” in *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2013, pp. 1735–1739. DOI: 10.7873/DATE.2013.349.

[11] W. Zimmermann and R. Schmidgall, *Busssysteme in der Fahrzeugtechnik [Bus systems in automotive engineering]*, ger. Springer Vieweg, 2014, p. 96.

[12] G. Macke, U. Hanke, W. Raether, and P. Viehmann-Schweizer, *Kompetenzorientierte Hochschuldidaktik [Competence-oriented university didactics]*, ger, 3rd ed. Beltz Verlagsgruppe, 2016, ISBN: 9783407294852. [Online]. Available: https://content-select.com/de/portal/media/view/56cc0a3a-741c-4bd7-8eab-5eeeb0dd2d03 (retrieved: 03/2022).