

Security Challenges in Automotive Hardware/ Software Architecture Design

Florian Sagstetter¹, Martin Lukasiewicz¹, Sebastian Steinhorst¹,
Marko Wolf², Alexandre Bouard³, William R. Harris⁴, Somesh Jha⁴,
Thomas Peyrin⁵, Axel Poschmann⁵, Samarjit Chakraborty⁶

¹ TUM CREATE, Singapore, ² escrypt-Embedded Security GmbH, Germany,

³ BMW Group, Germany, ⁴ University of Wisconsin, Madison, USA

⁵ Nanyang Technological University, Singapore, ⁶ TU Munich, Germany

Abstract—This paper is an introduction to security challenges for the design of automotive hardware/software architectures. State-of-the-art automotive architectures are highly heterogeneous and complex systems that rely on distributed functions based on electronics and software. As cars are getting more connected with their environment, the vulnerability to attacks is rapidly growing. Examples for such wireless communication are keyless entry systems, WiFi, or Bluetooth. Despite this increasing vulnerability, the design of automotive architectures is still mainly driven by safety and cost issues rather than security. In this paper, we present potential threats and vulnerabilities, and outline upcoming security challenges in automotive architectures. In particular, we discuss the challenges arising in electric vehicles, like the vulnerability to attacks involving tampering with the battery safety. Finally, we discuss future automotive architectures based on Ethernet/IP and how formal verification methods might be used to increase their security.

I. INTRODUCTION

In modern cars, innovations are mainly driven by electronics and software. As a result, top-of-the-range vehicles comprise up to 100 Electronic Control Units (ECUs) and multiple heterogeneous buses connected via gateways. Wireless communication, like keyless entry systems or WiFi, connect the car with its surroundings while functionality in upcoming cars will be even more based on software with strong wireless connectivity. Similar to the first computers connected to the Internet, current automotive architectures have not been designed for security, making them highly vulnerable to attacks aiming at gaining access to the system. Recently, a security analysis of a production vehicle revealed that an attacker might tamper with the brakes while the car is being driven [1], after gaining access to the in-vehicle network via Bluetooth or 3G [2]. Furthermore, car-thieves have been exploiting security breaches in the keyless entry system [3], or generate spare keys using the on-board diagnosis system to steal a car [4]. The lack of security measures in today's vehicles so far only causes some, mostly financial, damages to different parties, e.g., through spurious warranty claims after illegal chip tuning or mileage manipulations. However, without a significant change of the design paradigm of automotive systems to increase the vehicle security, cyber-terrorism attacks addressing vehicles

are only a question of time and inadequate security will become a severe safety issue.

Organization of the paper. The remainder of the paper is organized as follows. First, in Section II, an introduction is given to the history of automotive architectures, as well as threats that arise. Furthermore, we outline the challenges arising when security is embedded into automotive architectures and discuss current approaches. Section III presents an overview of security threats arising for electric vehicles. This includes tampering with the battery management system or the charging plug as an intrusion point. The security challenges of current automotive architectures might be overcome by next-generation in-vehicle networks and formal verification. Therefore, Section IV discusses future automotive architectures based on Ethernet/IP, before outlining how formal verification might be used to avoid vulnerabilities already during the design process. Finally, Section V makes concluding remarks.

II. THREATS AND INITIAL SECURITY SOLUTIONS

This section first gives an overview of the history of automotive security, before discussing threats for automotive architectures. Finally, we discuss the challenges arising when security is embedded into automotive architectures and discuss current approaches.

A. History of Automotive Security

Until 20 years ago, automotive security was restricted to mechanic car keys as well as alarm devices and mechanical (steering wheel) locks, protecting vehicles against theft and unauthorized usage. Automotive attacks were limited to car-theft and, rather seldom, manipulations of mechanical odometers and truck tachograph devices.

However, with the introduction of the first remote car keys, mandatory electronic diagnosis interfaces (e.g., On Board Diagnostics port) and the first on-board computers in the early 1990s, the situation has changed considerably. Since then, closed, mechanical car systems have changed into complex, digitally networked, and software-based IT systems. In the beginning, rather simple electronic tools allowed manipulating

TABLE I
LISTING OF DIFFERENT ATTACKER TYPES, THEIR TECHNICAL KNOWLEDGE, THE ACCESS TO THE NETWORK, AND THEIR GOAL.

attacker	technical knowledge	access	goal
car-thief	varied	wireless/physical	steal car
hacker	medium - high	wireless	fame
criminal	medium - very high	wireless/physical	harm passengers
workshop/tuner	medium - very high	physical	modify settings
counterfeiter/ competitor	high - very high	physical	study architecture

(digital) odometers or do illegal chip tuning without leaving visible traces. As a consequence, the automotive industry introduced more sophisticated automotive security solutions for keyless entry systems, electronic immobilizers, and vehicular component protection based on first applications of modern cryptography.

Today, vehicles use powerful digital infotainment or distributed safety functions, comprising up to 100 million lines of code [5]. At the same time, wireless interfaces connect the car with its surroundings, turning the vehicle into a nearly 24 hours online Internet node. Automotive attackers today are not only limited to car-thieves, and garage employees illegally modifying the functionality of a car (e.g., chip tuning, mileage manipulation). With the wireless connectivity of today's vehicles, hacking attacks to obtain information about the passengers become possible (e.g., location tracking, eavesdropping of communication) [2], as well as criminals might exploit this security flaws to harm passengers [1]. At the same time, very powerful criminal organizations (e.g., selling counterfeits, attacking after-market business models), or concurring manufactures (e.g., industry espionage) have a great interest in gaining information about an automotive architecture or modifying it. Table I outlines the different attackers and their goals.

However, in contrast to some years ago, today no automotive manufacturer denies the new IT security threats anymore. For instance, the AUTomotive Open System ARchitecture (AUTOSAR) standard which all major car manufacturers and suppliers follow [6] already defines security features. Furthermore, industry projects like E-safety vehicle Intrusion proTected Applications (EVITA) [7], or Security in Embedded IP-based Systems (SEIS) [8] have defined secure architectures for next-generation vehicles which might significantly improve the security of automotive architectures. However, various questions have not been answered yet, and further efforts are necessary to ensure that future functions like Car2X (C2X) do not become a threat to passenger safety, due to security issues.

B. Attack scenarios

This section presents a selection of potential vulnerabilities in a car. We first give an introduction to the general structure of automotive architectures, before discussing potential threats.

In-vehicle network structure. A vehicle today integrates a heterogeneous network of distributed ECUs. The ECUs

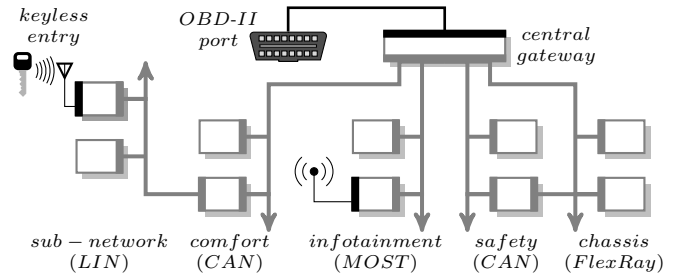







Fig. 1. Schematic illustration of a typical in-vehicle network architecture of a modern automobile. The different ECUs (□) are interconnected via different buses. Gateways are used to interconnect the buses with each other (◻) as well as to connect the network with the OBD-II diagnosis bus and for wireless access (◻).

communicate over different buses and protocols ranging from the low speed bus Local Interconnect Network (LIN) [9] over Control Area Network (CAN) [10] to fast buses like FlexRay [11] or Ethernet. The various buses are interconnected by gateways, creating a fully connected network as illustrated in Fig. 1. For maintenance and diagnosis, a legally mandatory On Board Diagnostics (OBD-II) port is installed under the dashboard of all new vehicles. It allows reading and writing data from and to the in-vehicle network and installing software on ECUs. European regulations require car manufacturers to make the information required to access the OBD-II port available to independent workshops for maintenance. Hence, tools and knowledge to access the in-vehicle network are widely available. From a security perspective, the OBD-II port is one of the most vulnerable attack points, as it gives the attacker full access to all ECUs. However, as the in-vehicle network is fully connected, any hi-jacked ECU might manipulate the in-vehicle network and the ECUs connected to it. Therefore, any ECU accessible from outside the vehicle provides a potential intrusion point, including particularly the wireless access points. Table II gives an overview over potential intrusion points for an attacker.

Wireless access to car. Modern cars are equipped with a variety of wireless communication protocols like 3G or Bluetooth, keyless entry systems, or Tire Pressure Monitoring Systems (TPMS). While all these functions provide improved comfort, weaknesses in specific implementations have been demonstrated for all of these protocols. For instance, in [2] the authors were able to exploit vulnerabilities in 3G as well as Bluetooth to gain access to the in-vehicle network. Weak security measures in keyless entry systems, allow to unlock a car [3] without the car key, and TPMS might allow data readout [12].

Physical access to car. Further threats arise if the attacker has physical access to the car which is the case in thefts. The previously mentioned OBD-II port provides various functions to the user which might be exploited by car-thieves. In recent years, several reports have been released about thieves gaining access to the OBD-II port to program a new key by fooling the

TABLE II
SELECTION OF POTENTIAL INTRUSION POINTS FOR TODAY'S AUTOMOTIVE ARCHITECTURES.

	intrusion point	distance to car/ access point
wireless access		3G long-range
		WiFi short-range
		Bluetooth, remote key near-field
physical access		OBD-II diagnosis port giving access to in-vehicle network
		unmounted ECU direct hardware access

anti-theft systems. Common ways to gain access to the vehicle interior is exploiting blind spots in the intrusion detection or even preventing the car owner from locking the car through Radio Frequency (RF) jammers which overlay the locking command with noise. See [4] for reports on such thefts.

Access to in-vehicle network. Once an attacker has obtained access to the in-vehicle network, additional security measures like authentication are required to protect the network nodes. However, as demonstrated in [1], a lack of such measures is not uncommon. The vulnerability of the network strongly depends on the bus type a tampered ECU is connected to. For instance, FlexRay or LIN require a predefined schedule which exactly defines at which time each node is allowed to send messages. This strongly limits the communication possibilities of a tampered ECU without gateway functionality. In contrast, the CAN bus allows adding new participants to the network in a plug and play manner, transmitting messages based on fixed priorities. For instance, from a tampered ECU, an attacker could easily inject a large number of messages with a high priority, hindering the correct functionality of other functions without having any knowledge about the architecture. For a more detailed overview about security issues of different buses, see [13], [14], [15].

C. Limitations of automotive networks and current security solutions

The majority of the ECUs in a vehicle possess only limited computation power and limited memory resources. For instance, low-end ECUs might only be 8bit microcontrollers running at 20Mhz with 32kB memory and 1kB of RAM. This strongly limits their ability to perform cryptographic operations like message encryption for real-time functions. Additionally, the predominant CAN bus or the LIN bus only support 8 byte messages which does not allow appending data segments as required for message authentication. Hence, several automotive buses in use today are not suitable for secure in-vehicle communication between ECUs.

Current security solutions. Vehicles today already implement some basic security which is not only limited to immobilizers or wireless communication modules. Critical ECUs are protected by both cryptographic functions as well as physical security to prevent modifications. For instance, an

authentication is commonly required before a firmware update process can be initiated. However, this authentication process might be weak and does not provide sufficient protection [1]. Modern gateways, connecting different buses, allow message filtering which might prevent malicious messages from being sent in the in-vehicle network. Finally, to overcome the limitations of automotive ECUs, microcontroller manufacturers equip their latest generation ECUs with cryptographic modules like Secure Hardware Extension (SHE) [16] which allows an authenticated boot process to prevent software manipulations. While these efforts provide some basic security to single components, they only allow a partial protection for automotive architectures. This is especially critical, as attackers generally do not attack secure components rather than the vulnerabilities in their integration. Therefore, only a paradigm-shift in the design of automotive architectures to a holistic design approach, taking security into account from the beginning, allows creating a secure architecture.

III. SECURITY FOR ELECTRIC VEHICLES

This section gives an overview of additional security vulnerabilities of electric vehicles and how they are currently addressed. While electric vehicles have many security vulnerabilities in common with combustion engine cars, we see three additional security threats arising: (1) The battery which might ignite a fire when damaged, (2) the charging plug as an additional intrusion point, and (3) the upcoming drive-by-wire functionality which might be exploited to maliciously control a car.

A. Battery security

As recent reports on electric vehicle batteries catching fire indicate, vehicle batteries might be targeted by an attacker to harm passengers. Batteries for electric vehicles generally consist of various single cells which are controlled by a central Battery Management System (BMS) [17]. The BMS monitors the cell voltages and temperature, and also controls the current flow to and from the battery, including the charging strategies. It is therefore responsible to ensure correct operation and prevent damage.

Controlling the BMS allows an attacker to control all battery functions, including ignoring critical battery conditions and tolerating too high voltages and currents to damage the battery. This might allow an attacker to severely damage a battery and even to ignite a fire. However, for safety reasons, modern batteries employ pressure release valves or burst open to reduce pressure and prevent combustion [18]. Additionally, many BMSs implement hardware watchdog functionality which disconnects battery cells if certain voltage or temperature ranges are exceeded. Nevertheless, while this safety functionalities might prevent fire ignition, an attacker might still be able to irreparably damage the battery or harm the passenger. For instance, the BMS might disconnect the battery from the engine during acceleration. Therefore, during the design phase of an in-vehicle network, particular care needs to be put in protecting the BMS against malicious attacks.

Another security threat arising are counterfeit batteries. While this primarily leads to financial damage for the car manufacturers and battery suppliers, e.g., through spurious warranty claims, counterfeit batteries also provide a huge security risk, as they might lack safety mechanisms and are not certified. Therefore, it is essential to establish an authentication mechanism for batteries which allows the BMS to verify that only original batteries are installed.

B. The charging plug as intrusion point

An essential part of a full electric vehicle is the charging plug for recharging the battery. While the charging plug used to be a simple electric plug for the first generations of electric vehicles, today various standards exist which also implement a communication protocol to allow information exchange between the BMS and the charging station. For instance, the CHAdeMO standard [19], widely used in Japan, relies on a CAN bus connection to the vehicle for the communication while the IEC 61851 standard [20] used in Europe relies on a power line communication. In particular, a communication over the CAN protocol bares high risks if it is directly connected to the in-vehicle network without message filtering as it is not uncommon for legacy CAN-CAN gateways in today's vehicles. This might allow a thief to program a new key to unlock the car or criminals to reprogram ECUs through the charging plug. Plans for the next generation charging plugs include additional services like multimedia streaming or even firmware updates [20] which further increases the risk of attacks. This is particularly critical as a communication over the charging plug is highly vulnerable to man-in-the-middle attacks where the attacker might attach a connector between the charging plug of the car and the charging station. This would allow an attacker to eavesdrop the communication or to modify packages.

To protect the vehicle against attacks through the charging plug, the upcoming ISO 15118 standard suggests the Transport Layer Security (TLS) protocol [20], [21]. TLS would then provide the required security for a communication over the charging plug for future vehicles. Additionally, security measures are required to prevent manipulation of energy charging payment or privacy issues, see [22] for a detailed discussion.

C. Drive-by-wire functionality and arising threats

While drive-by-wire is a technology emerging also for combustion engine cars, it is of particular importance for electric vehicles. As current battery technology only allows a limited range for electric vehicles, energy recuperation during braking is essential to extend the driving range [24]. Depending on the situation, conventional brakes are required to support the deceleration for emergency braking as illustrated in Fig. 2. If the vehicle brakes are mechanically connected to the braking pedal, only partial energy recuperation is possible. Therefore, a mechanical decoupling between the braking pedal and the brakes becomes necessary as provided by brake-by-wire [25].

While drive-by-wire provides various benefits, it also leads to great security risks as it would theoretically allow remotely

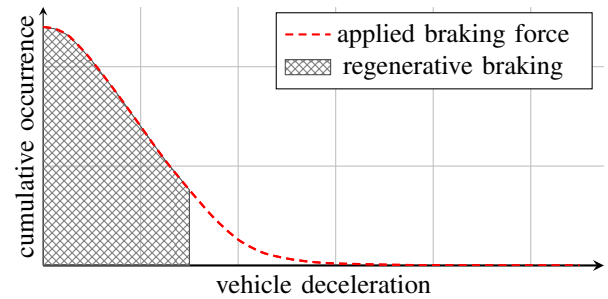


Fig. 2. Illustration of a typical braking maneuver distribution during the life-time of a vehicle [23]. While for electric vehicles energy recuperation is not sufficient for emergency braking, it might theoretically be applied for more than 85% of all braking maneuvers and support conventional brakes for all other maneuvers.

controlling the vehicle or deactivating the brakes. Hence, to prevent the misuse of drive-by-wire by attackers, it is necessary to establish a secure in-vehicle network which is both isolated from its surroundings and supports a secure and authenticated in-vehicle communication.

IV. FUTURE IN-VEHICLE NETWORKS: A SECURITY ORIENTED VIEW

This section first presents a potential solution for next-generation in-vehicle networks based on Ethernet/IP, before discussing how formal verification might be used to increase vehicle security.

A. Networking Solutions for the Future

Future applications for driver assistance, infotainment and external connectivity have all increasing requirements in bandwidth availability. But security concerns and technology limitations in term of bandwidth and interoperability currently constrain their development and integration in cars. Part of the solution may reside in the use of buses supporting a higher bandwidth in combination with more flexible networking protocols like Ethernet/IP, an option already investigated by the industry project SEIS [26] for several reasons: (1) Limited cost, through inexpensive single pair unshielded cables. (2) Larger bandwidth, as the automotive variant of the 100 Mbit Ethernet would multiply the current bandwidth capacity by ten and may soon lead to its Gbit version. (3) Scalable and easy ECU coupling, as automotive switches would simplify the network addressing, supporting unicast/multicast communication. (4) Available standards, like many standard Internet protocols, would be directly applicable for the automotive purpose. While being functionally suitable, Ethernet/IP does not directly solve every security issue. The rest of this section presents some considerations related to the on-board security management, the integration of external use cases and some migration challenges.

Ethernet/IP and on-board security. With IP being a well-known standard which is in use for several years, it has proven to be secure [27]. Protocols, like IPsec/IKEv2 or TLS have been strengthened over the years and are mature

enough for an automotive use case as well. During the SEIS project, a security architecture for an on-board network was defined which features a domain-based allocation of the ECUs depending on their purpose (e.g., infotainment or power train management). Master-ECUs, located at the entry of each domain network, enforce filtering and network-based intrusion detection, while the other ECUs are in charge of setting up their own secure communication channels over IPsec. Every ECU disposes of a engineering-driven communication middleware abstracting both security and network management [28]. Developers might then focus on the application logic, while network addressing, choice of security protocol and policy management are automatically performed within the middleware. Several middleware versions with different security levels might then allow coping with every use case and its requirements and can integrate additional hardware-based mechanisms (e.g., secure key storage, remote platform attestation) enhancing the communication security.

Security and external mobility services. IP allows simpler in-car integration of car-to-X (C2X) use cases like cloud computing, smartphones, or loadable third-party applications and therefore increases the car threat level. The larger bandwidth not only provides a functional advantage, it also allows exchanging additional security metadata to enforce a complex C2X authorization model and system monitoring [29]. In addition, if most of the external communication interfaces (e.g., LTE, Wi-Fi) would be centralization around a multi-platform antenna [30], the car manufacturers would have the opportunity to build a central C2X security gateway allowing easy maintenance and simplified security verifications.

Migration towards Ethernet/IP. Ethernet/IP based communication provides a reliable basis for automotive innovation. The provided security protocols and bandwidth may allow to reach a holistic security solution. Though automotive systems are complex and include multiple electronic components, whose designs and implementations will always involve several actors from diverse companies. Several standardization committees for an automotive Ethernet or IP-based middleware have already started and security should follow soon. Besides a partial transition to Ethernet/IP in cars is already planned for 2018 and foresees the cohabitation with other traditional bus technologies [31]. While providing a progressive migration, this cohabitation will let part of the system unprotected. New and complementary security mechanisms will be required for both non-IP- and IP-enabled systems in order to detect ongoing attacks and avoid critical functionalities to get compromised.

B. Formal Methods for Vehicle Security

This section gives an introduction into formal verification of security properties and discusses an application to automotive architectures. In today's automotive architectures, many applications are developed by suppliers and the component integration requires a significant amount of time of the development of a vehicle. The integration process is still mainly performed

manually and requires intensive testing while being highly error-prone. A model-based design approach, in combination with formalized verification methods, would significantly reduce the testing and integration efforts through verifying the correct functionality and thereby security.

Hence, formal methods for verifying the correctness of transition systems can be extended and applied to check the security of vehicles. In general, formal methods are applied to specify the correct behavior of a system, and either prove that the system satisfies its specification, or construct a system behavior that violates the specification. In the context of vehicle security, formal methods might be applied to specify the secure behavior of a vehicle, and either determine that a vehicle is always secure, or produce an attack that causes the vehicle to exhibit an insecure behavior.

Future automotive architectures, which have been designed for security, will consist of components that enable a formal verification of security aspects. While, in general, formal verification of automotive embedded software suffers from complexity challenges, an approach could be to *design for verifiability*. Such a design methodology would implement critical functions on a micro kernel, for which a formal modeling and, hence, formal verification is possible. For such architectures, one could apply techniques from the domain of *assume-guarantee reasoning* [32], in which the system is viewed as a composition of a set of components. A system designer provides both properties that need to hold in the execution environment of each component, and guarantees that the designer believes to hold for the results of each component, as long as the component's assumptions hold. The designer then verifies that (1) the assumptions and guarantees of each component imply that the system composed from the components satisfies the overall definition of correctness, and (2) that if each assumption holds, each component does indeed fulfill its guarantee. However, verification can only be successfully applied if an accurate model and definition of assumptions have been defined. In particular, a system designer must take great care to provide a set of assumptions that are weak enough that they model real-world threats, yet are strong enough that they allow the system to be verified.

Once a system designer has determined a behavior violating the correctness through the application of formal methods, the developer must then redesign, or *patch*, the system such that it does not demonstrate the violation. As these security flaws might be determined during the integration process, no knowledge about the source code of software that runs on the component might be given. To aid designers in patching such systems, for next generation vehicles, techniques that automatically patch the embedded software that runs on vehicle components might be applied.

Previous work has addressed how to automatically patch software that runs on general-purpose computing platforms [33]. However, automatically patching the embedded software in vehicles poses new challenges. Specifically, a critical step to automatically patching a system is to take a specific attack on a system and infer the general vulnerability

in the system that allows the attack (i.e., the *root cause* of the attack). Determining the root cause of an attack on a vehicle is particularly challenging because the attack may exploit behaviors of multiple components of the vehicle.

V. CONCLUDING REMARKS

This paper gave an introduction to the security of software and hardware architectures of automobiles. We present threats for automotive architectures and the challenges arising when security is embedded into the vehicle architecture. Furthermore, we give an introduction into the security of Electric Vehicles and discuss a future automotive architecture based on Ethernet/IP.

Embedding security into a car is a challenging task as the security of a vehicle needs to be ensured over the whole life-span of a car with 15 years and more. While wireless communication protocols are already connecting the car with its surroundings, upcoming technologies like C2X or the appstore rise security questions which have not been satisfactorily answered. In addition, electric vehicles introduce further security questions which might not be answered by a holistic security approach, but rather require an independent solution. The Ethernet/IP based on-board network under development by the automotive industry in combination with a middleware and message filtering might form the basis for a secure in-vehicle network. However, various security issues are not resolved yet and require additional solutions.

Cars today consist of various components from different suppliers which are integrated into one system. Integrating all these components into a secure architecture is almost impossible, as generally little is known about the supplier hardware. However, for a secure architecture, a holistic design approach is necessary which takes the correlation of different components into account. A model driven design approach in combination with formal verification would allow verifying the security of an automotive architecture already during the design process and avoid security flaws from an early design stage on.

REFERENCES

- [1] 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 *IEEE Symposium on Security and Privacy*, 2010, pp. 447–462.
- [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. of USENIX Security*, 2011.
- [3] S. C. Bono, M. Green, A. D. Rubin, M. Szydlo, and A. Juels, "Security Analysis of a Cryptographically-Enabled RFID Device," in *Proc. of USENIX Security*, 2005.
- [4] The Telegraph, "Thieves placed bugs and hacked on-board computers of luxury cars," 2012. [Online]. Available: <http://www.telegraph.co.uk/news/uknews/crime/9369783/Thieves-placed-bugs-and-hacked-onboard-computers-of-luxury-cars.html>
- [5] R. Charette, "This car runs on code," *IEEE Spectrum*, vol. 46, no. 3, p. 3, 2009.
- [6] AUTOSAR, "AUTOSAR 4.0," 2010. [Online]. Available: <http://www.autosar.org>
- [7] M. Wolf and T. Gendrullis, "Design, Implementation, and Evaluation of a Vehicular Hardware Security Module," in *ICISC 2011*, 2011.
- [8] "Sicherheit in Eingebetteten IP-basierten Systemen (in German)." [Online]. Available: <http://www.strategiekreis-elektromobilitaet.de/public/projekte/seis>
- [9] LIN Consortium, "LIN specification package V2.2A." [Online]. Available: <http://www.lin-subbus.org/>
- [10] CAN, "Controller Area Network." [Online]. Available: <http://www.can.bosch.com>
- [11] FlexRay Consortium, "FlexRay Communications System Protocol Specification version 3.0.1," 2010. [Online]. Available: <http://www.flexray.com>
- [12] A. Wright, "Hacking cars," *Communications of the ACM*, vol. 54, no. 11, p. 18, Nov. 2011.
- [13] M. Wolf, A. Weimerskirch, and C. Paar, "Security in Automotive Bus Systems," in *Proc. of ESCAR*, 2004.
- [14] D. Nilsson, U. Larson, F. Picasso, and E. Jonsson, "A First Simulation of Attacks in the Automotive Network Communications Protocol FlexRay," in *Proc. of CISIS*, vol. 53. Springer, 2009, pp. 84–91.
- [15] T. Hoppe, S. Kiltz, and J. Dittmann, "Security Threats to Automotive CAN Networks Practical Examples and Selected Short-Term Countermeasures," *Computer Safety, Reliability, and Security*, vol. 5219, pp. 235–248, 2008.
- [16] Hersteller Initiative Software, "SHE Secure Hardware Extension V1.1," 2009. [Online]. Available: <http://www.automotive-his.de>
- [17] M. Brandl, H. Gall, and M. Wenger, "Batteries and battery management systems for electric vehicles," *Proc. of DATE*, pp. 971–976, 2012.
- [18] C. Mikolajczak, M. Kahn, K. White, and R. Long, *Lithium-Ion Batteries Hazard and Use Assessment*. Springer, 2012.
- [19] CHAdEMO Association, "CHAdEMO," 2010. [Online]. Available: <http://www.chademo.com/>
- [20] R. Falk and S. Fries, "Electric Vehicle Charging Infrastructure Security Considerations and Approaches," in *Proc. of INTERNET*, 2012, pp. 58–64.
- [21] T. Dierks and E. Rescorla, "RFC5246 - The Transport Layer Security (TLS) Protocol Version 1.2," IETF, Tech. Rep., 2008.
- [22] C. Paar, A. Rupp, K. Schramm, A. Weimerskirch, and M. Wolf, "Implementing Data Security and Privacy in Next- Generation Electric Vehicle Systems," *SAE Technical Paper 2010-01-0743*, 2010.
- [23] B. Heissing and M. Ersoy, *Chassis Handbook: Fundamentals, Driving Dynamics, Components, Mechatronics, Perspectives*. Vieweg+ Teubner Verlag, 2010.
- [24] C. C. Chan, "The State of the Art of Electric, Hybrid, and Fuel Cell Vehicles," *Proceedings of the IEEE*, vol. 95, no. 4, pp. 704–718, Apr. 2007.
- [25] E. Bretz, "By-wire cars turn the corner," *IEEE Spectrum*, vol. 38, no. 4, pp. 68–73, 2001.
- [26] M. Glass, D. Herrscher, H. Meier, and M. Piastowski, "Seis - security in embedded ip-based systems," *ATZelektronik worldwide*, no. 2010-01, pp. 36–40, 2010.
- [27] NIST, "National vulnerability database," <http://web.nvd.nist.gov/view/vuln/statistics>.
- [28] A. Bouard, B. Glas, A. Jentsch, A. Kiening, T. Kittel, F. Stadler, and B. Weyl, "Driving automotive middleware towards a secure ip-based future," in *Proc. of ESCAR*, 2012.
- [29] A. Bouard, J. Schanda, D. Herrscher, and E. Eckert, "Automotive proxy-based security architecture for ce device integration," in *Proc. of Mobileware*, 2012.
- [30] C. F. Mecklenbrauker, A. F. Molisch, J. Karedal, F. Tufvesson, A. Paier, L. Bernado, T. Zemen, O. Klemp, and N. Czink, "Vehicular channel characterization and its implications for wireless system design and performance," vol. 99, no. 7, 2011.
- [31] A. Maier, "Ethernet - the standard for in-car communication," In *2nd Ethernet & IP @ Automotive Technology Day*, 2012.
- [32] E. M. Clarke, O. Grumberg, and D. Peled, *Model checking*. MIT Press, 2001.
- [33] D. Brumley, J. Newsome, D. X. Song, H. Wang, and S. Jha, "Towards automatic generation of vulnerability-based signatures," in *IEEE S&P*, 2006.