2011 IEEE Intelligent Vehicles Symposium (IV)
Baden-Baden, Germany, June 5-9, 2011

Security Aspects of the In-Vehicle Network in the Connected Car

Pierre Kleberger, Tomas Olovsson, and Erland Jonsson
Department of Computer Science and Engineering
Chalmers University of Technology
SE–412 96 Gothenburg, Sweden
Email: {pierre.kleberger,tomas.olovsson,erland.jonsson}@chalmers.se

***Abstract*—In this paper, we briefly survey the research with respect to the security of the connected car, and in particular its in-vehicle network. The aim is to highlight the current state of the research; which are the problems found, and what solutions have been suggested. We have structured our investigation by categorizing the research into the following five categories: problems in the in-vehicle network, architectural security features, intrusion detection systems, honeypots, and threats and attacks. We conclude that even though quite some effort has already been expended in the area, most of it has been directed towards problem definition and not so much towards security solutions. We also highlight a few areas that we believe are of immediate concern.**

I. Introduction

This paper surveys current research in securing the in-vehicle network of the connected car. The aim is to highlight current research within the area, so that new directions can be taken in the future.

Equipping the vehicle with a wireless connection will create many opportunities for new services, e.g. firmware update over the air (FOTA) [1] and remote diagnostics [2]. However, those very attractive features come with great challenges. The internal as well as external communication must be properly secured. This is particularly important, since the in-vehicle network is safety-critical and it is imperative to avoid that security problems lead to disastrous safety implications. Therefore, we are concerned about some recent papers that report about significant insecurity of the connected car.

For example, the lack of security in today's vehicles was just recently shown by Koscher et al. [3]. By connecting to the On-Board Diagnostics II (OBD-II), they were able to, among other things, issue commands to disable the breaks while driving. Although these attacks were performed through the diagnostic interface, which so far requires physical access to the vehicle, we expect that these attacks will be possible to perform through a wireless connection in a coming version of the connected car.

Some systems within the vehicle have been designed with security in mind, such as the electronic immobilizer [4], but most have not and a complete security architecture is yet to be defined. As a further complication, there is reason to believe that the introduction of Vehicle–to–Vehicle (V2V) and Vehicle–to–Infrastructure (V2I) communication will present an even higher threat level and as a consequence the security requirements will be increased accordingly.

The rest of this paper is outlined as follows. Section II presents the related research within the area. In Section III, we give a background to the vehicle settings. Section IV presents the research found in securing the in-vehicle network of the connected car. The paper concludes with a discussion in Section V including open research issues followed by conclusions in Section VI.

II. Related Work

A few surveys and overviews of the security within the connected car have been published earlier. However, in this paper we focus on the security of the in-vehicle network and we are not aware of any other work with this focus.

Wolf et al. [4] survey the security within the vehicle. Possible attacks, protection mechanisms, and some security-critical applications are presented.

Jenkins and Mahmud [5] discuss security problems and attacks towards the vehicle. They look at inter-vehicle and in-vehicle communications, and also at software and hardware attacks. A further introduction to security for embedded systems is given by Kocher et al. [6].

Brooks et al. [7] show with use-cases what needs to be protected in a vehicle and different scenarios of what operations may be conducted on the vehicle. The possible communication means to the vehicle were also classified. They further use an adapted version of the CERT Taxonomy [8] to analyse attacks towards services already implemented in the vehicle or that will come in the near future. Among the services analysed were the need for secure update of firmware in Electronic Control Units (ECUs) and attack risks when the vehicle becomes more and more integrated into the systems of the automotive company. One example of such a system is remote diagnostics.

A defence-in-depth approach for securing the vehicle is discussed by Larson and Nilsson [9]. The five layers they look at are; prevention, detection, deflection, countermeasures, and recovery. For the five layers, they also discuss the possible needs; authentication to prevent unauthorized access, Intrusion Detection System (IDS) and logging mechanisms for detection, suggestion of using honeypots for information

978-1-4577-0891-6/11/$26.00 ©2011 IEEE

retrieval, Intrusion Prevention System (IPS) as a countermeasure, and the necessity of traceability to perform recovery. In [10], Nilsson and Larson extend the discussion and present their approaches for the different layers.

There are only a few research projects where the main focus is to secure the connected car or the communication with it. Two of them are EVITA [11] and SeVeCOM [12].

III. Background

A. The Connected Car

The connected car consists of three domains [13]:

(1) the *vehicle*, consisting of the in-vehicle network and ECUs,
(2) the *portal* at the automotive company, delivering services to the vehicle, and
(3) the *communication link* between the vehicle and the portal.

The in-vehicle network can further be divided into sub-networks of different bus system technologies; *Controller Area Network (CAN)*, *Local Interconnect Network (LIN)*, *Media Oriented Systems Transport (MOST)*, and *FlexRay*. The sub-networks are connected to each other through special *gateway ECUs*.

Of the three domains above, we will focus on the vehicle.

B. Challenges

There are some general requirements that present special challenges for securing the in-vehicle network:

(1) resource constrains of the ECU, i.e. the ECU has limited processing power and memory.
(2) limited possibilities of extra cost for the connected devices, new security solutions must be very cost efficient.
(3) lifetime of the solution, the vehicle may be in use for 10-15 years.

C. Attacker Model

There have been different approaches in using attacker models when addressing the security for the vehicle settings. One approach has been to not really define or use an attacker model [3, 14]. In [3], Koscher et al. assume they have necessary access to the in-vehicle network to perform their attacks.

Another approach has been to derive an attacker model from the CERT Taxonomy proposed by Howard and Longstaff [8]. One such model for traffic on the CAN-bus is derived by Nilsson and Larson [15], where the attacker can read, spoof, drop, modify, flood, steal, and replay traffic. The model was further applied to the FlexRay-protocol [16].

In [17], Lang et al. use an attacker model based on IP traffic, where the attacker can read, modify, interrupt, create/spoof, and steal/remove traffic.

IV. In-Vehicle Network

In this chapter we present the research related to the in-vehicle network. We first highlight current problems found within the in-vehicle network. We then look at architectures proposed to implement security into these networks. Proposals of using IDSs, and honeypots are then covered, followed by research on threats and attacks. Figure 1 shows the taxonomy.

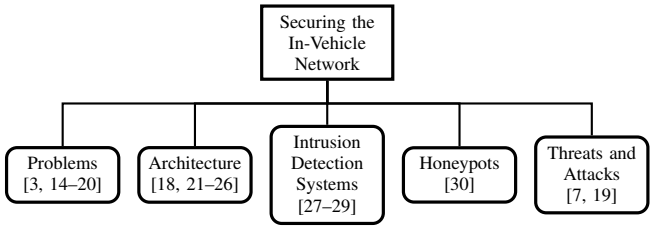

Fig. 1: Taxonomy of In-Vehicle Network

A. Problems in In-Vehicle Networks

Most of the work in addressing security of the in-vehicle network has been towards identifying and showing on the lack of security. In this section we will first present the related work being done in the area and then summarize the problems that have been identified.

1) Related work: Koscher et al. [3] have recently highlighted that there is a significant lack of necessary security mechanisms in in-vehicle networks. They conducted experiments on two vehicles. Using techniques such as packet sniffing, packet fuzzing, and reverse-engineering, they found a number of attacks that could be performed towards the in-vehicle network.

Wolf et al. [18] have investigated some possible attacks towards different buses in the in-vehicle network.

Hoppe and Dittmann [19] used simulations for evaluating security. They investigated the possibility of performing sniffing and replay attacks on the CAN-bus by simulation of an electronic window lift system. To classify their attacks, an adapted version of the CERT Taxonomy proposed by Howard and Longstaff [8] was used. In [14], Hoppe et al. also performed attacks towards the electronic window system using real hardware as well as attacks towards the warning lights of the anti-theft system and the air-bag control system.

Nilsson and Larson [15] introduced the concept of a vehicle virus. The virus was listening for the message on the CAN-bus that locks the doors remotely, and when that message was captured, the virus took appropriate actions. To further address the needs of classifying techniques in protecting against vehicle viruses, an adapted version of the taxonomy by Hamle and Bauer [31] is proposed.

Finally, Lang et al. [17] provide an interesting discussion of the security implications when the vehicle is connected using an IP-based network. Nine ”hypothetical attack scenarios” were suggested based on attacks known from ”ordinary IT systems”, i.e. attacks on the communication protocols, malicious code, and social engineering. Each scenario was analysed with respect to confidentiality, integrity, availability, authenticity, and non-repudiation. Also, an attempt to quantitatively estimate the impact on safety was conducted. Thus, for each of the scenarios a Safety Integrity Level (SIL) value was proposed.

2) Identified security problems: Here follows a summary of the identified security problems:

- *lack of sufficient bus protection.* The CAN-bus lacks necessary protection to ensure confidentiality, integrity, availability, authenticity, and non-repudiation [14]. Messages on the CAN-bus can be read by other nodes, have no sender or receiver address, and are not protected by any Message Authentication Code (MAC) or digital signature. Analysis of the CAN and FlexRay specifications [15, 16] also concludes that these protocols lack necessary protection of data authentication, data confidentiality, and data freshness. However, there is some protection for data integrity and data availability.
- *weak authentication.* It is possible to illicitly reprogram ECUs with new firmware [3]. The reason for this is weak authentication and sometimes no authentication at all.
- *misuse of protocols.* Attacks towards the in-vehicle network can be performed by misusing well chosen mechanisms in the protocols [18]. Thus, for the LIN-bus, sending malicious sleep frames could disable the subnet. For the CAN protocol, a Denial-of-Service (DoS) attack may be carried out using the bus arbitration mechanism. By sending messages with the highest priority, no one else will be able to use the bus. Furthermore, well formed malicious error messages could be used to attack the fault detection mechanism implemented in CAN and FlexRay, so that the controllers will disconnect from the network.
- *poor protocol implementation.* In some cases the protocol implementation is such that it does not properly reflect the protocol standard [3]. For example, the standard specifies that it should not be possible to put the Engine Control Module (ECM) into programming mode while the vehicle is moving. Obviously, this is for safety reasons. However, in some implementations it is indeed possible to launch a command that would disable the CAN communication and put the ECU into programming mode despite the fact that the vehicle is moving.
- *information leakage.* An information leakage from the vehicle can be triggered by manipulating the diagnostic protocol, creating a potential privacy violation [20]. The information leakage was accomplished by sniffing an ordinary diagnostic session, and then replay a modified version of the traffic. Since the gateway is unable differ ordinary traffic from diagnostic traffic, both types of traffic will be forwarded by the gateway.

B. Architectural Security Features

In this section we present a number of security features proposed for the in-vehicle architecture or communication. A summary of features is given in Table I.

Wolf et al. [18] suggest ways to improve the security of the communication by requiring authentication of the controllers and by encrypting the communication. First, each controller has to be authenticated by the gateway by means of a certificate. After authentication, the controller will receive a symmetric encryption key that is shared with other authenticated controllers on that local network so as to make secret data exchange possible.

TABLE I: Summary of Architectural Security Features with respect to communication

Ref.	Confidentiality	Integrity	Authentication	Communication	Timing
[21]	✓			–	Real-Time
[22]		✓	✓	End-to-End	Delayed
[23]	✓		✓[1]	Group	Real-Time
[24]	✓	✓	✓	End-to-End	Real-Time
[25]		✓	✓	Group	Delayed[2]

[1] Authentication of ECUs within group, not individual messages
[2] Uses Time-Triggered Protocol (TTP)

In [21], Chávez et al. suggested using the security services of the OSI Reference Model (ISO 7489-2) for securing the CAN-protocol. The OSI model describes five security services, confidentiality, integrity, authentication, non-repudiation, and access control. According to this, they proposed that access control could be taken care of at higher layers in the protocol, that integrity could be enforced by using hash algorithms, and that confidentiality could be enforced by using RC4 encryption of the CAN-frames. The authors then evaluated the encrypting time for different sizes of payload. The remaining two OSI services, authentication and non-repudiation was not considered to be useful in this context.

Nilsson et al. [22] propose the use of a MAC for providing data integrity and data authentication in the CAN communication. To achieve this, a 128-bit key is shared between the two communicating ECUs. By using the KASUMI encryption algorithm in Cipher-Block Chaining Message Authentication Code (CBC-MAC), a 64-bit MAC can be produced. The MAC is calculated over four consecutive CAN-messages and the resulting MAC is divided into four 16-bit blocks and transmitted in the Cyclic Redundancy Code (CRC)-field of the next four CAN-messages. The protocol introduces a delay before the data integrity and data authentication can be verified. In total, eight messages are needed for the verification to be completed. Two of the remaining challenges with the protocol were that; (1) if the MAC calculation fails, the actual individual message that was wrong can not be identified, and (2) there is no protection against replay attacks.

Groll and Ruland [23] propose an architecture where they divide the communication into trusted groups. All ECUs within a trusted group share the same symmetric key to encrypt and decrypt the communication. A Key Distribution Centre (KDC) within the vehicle is used for creating and distributing the symmetric keys for these trusted groups. The trusted groups are defined by Access Control Lists (ACLs) and are signed by the automotive company. One ACL is kept at each ECU and defines the trusted groups that the ECU belongs to. To distribute the symmetric keys for communication within the trusted groups to an ECU, the ECU sends its ACL to the KDC. After the KDC has

verified the signature on the ACL, the KDC sends back the symmetric keys for those trusted groups defined by the ACL. To protect the distribution of the trusted group keys, asymmetric encryption is used between the ECU and the KDC. The asymmetric keys needed must also be signed by the automotive company.

Oguma et al. [24] propose an attestation-based security architecture. By applying a hash-function, verifying the integrity of the software in the ECU and comparing the result with a list of valid hashes, they want to verify that the ECU only runs genuine software. Only successfully validated ECUs will be able to exchange symmetric keys for further encrypted communication. The architecture they propose has three components; (1) a center outside the vehicle, (2) a master ECU within the vehicle, and (3) the other ECUs within the vehicle. The center stores the information regarding all vehicles, but the master ECU is also needed in each vehicle to do attestation, since the center might not always be reachable. The master ECU holds a list of the hash-values that are valid for the software running on the ECUs for that vehicle. Furthermore, instead of using asymmetric encryption within the vehicle, a Key Predistribution System (KPS) is used. After the attestation process has been performed, encryption keys are generated for each pair of validated ECU using the KPS. Both encrypted messages and signed messages are supported. These messages also hold information to prove that the ECU has been validated and a counter to protect against replay attacks. In [32], Lee et al. further discuss the attestation-based security architecture. By using ProVerif, they propose a way to formally verify the protocol with respect to some of the requirements that Wolf et al. [4] bring up to be important for secure communication.

An approach to provide authentication of messages for time-triggered applications is proposed by Szilagyi and Koopman [25]. Thus, a protocol was designed to be able to authenticate multiple destinations at the same time, which requires that each pair of communicating nodes share a symmetric encryption key. These keys are used for calculating the MAC over the messages for each destination. Each MAC is further stripped down to a few bits and concatenated to the end of the message. Since it is easier to forge a message with only a few bits of the MAC available, the authors propose that authentication is provided by successfully verifying the MAC over a set of messages. For the two types of messages investigated, state-changing messages and reactive control messages, an upper boundary of the probability of performing a successful attack is discussed. The proposed protocol also has protection towards replay attacks. The protocol is further discussed in [33], where an analysis with the help of simulated attacks is provided.

A more generic approach has been suggested by Schulze et al. [26]. This is accomplished by the introduction of a Data Management System (DMS); instead of letting all ECUs exchange data with each other, data is stored in specific nodes within the vehicle. By using a DMS for storing data, security mechanisms such as access control could be enforced on access and updates of data as well as concurrency control to ensure data integrity. The method also opens for the possibility to store a global state to a protective storage in the case of an accident. Three different approaches to deploying the DMS are investigated: a centralized approach, a distributed approach, and a hybrid approach, in which a DMS is deployed for each sub-network. The hybrid DMS approach is found to be the most attractive.

C. Intrusion Detection Systems

So far, the research on Intrusion Detection Systems (IDSs) has been targeting the CAN protocol. Both specification-based and anomaly-based detection methods have been addressed. Here follow the approaches proposed.

1) Specification-based detection: Larson et al. [27] propose and evaluate a specification-based IDS for the CAN 2.0 and CANopen 3.01 protocols. They conclude that, since these protocols lack information about the producer and consumer of messages, there is not enough information available for using network-based intrusion detection. Instead, they propose host-based detection, i.e. one detector is placed in each ECU. The incoming and outgoing network traffic can then be investigated based on information from the protocol stack and the object directory of the CAN-protocol at the expected ECU. For the detector in the ECU, security specifications for the communication protocol and the ECU behaviour can be developed. For the communication protocol, the security specification is described by (1) requirements for individual fields, (2) one field's dependence of another field, and (3) an object's dependence of another object. Furthermore, the behaviour of the ECU is described by security specifications for (1) message transmission, (2) message retrieval, and (3) allowed rates for message transmission and retrieval.

From their evaluation of the specification-based approach, Larson et al. [27] conclude that the gateway ECU is the most important ECU to protect. If the gateway ECU is compromised, all attacks they investigated could be performed. Unfortunately, performing detection in the gateway ECU is harder than in ordinary ECUs, since the detectors for the different interfaces at the gateway have to cooperate to detect certain attacks, e.g. to detect lost or modified messages.

2) Anomaly-based detection: Hoppe et al. [29] demonstrate an anomaly-based IDS for the CAN protocol. In contrast to the specification-based approach by Larson et al. [27], where the IDS is placed in the ECU, they listen to the network traffic on the CAN-bus. By looking at the rate of how often specific messages are transmitted on the bus, and comparing that to what is considered to be normal, deviations of the number of transmitted messages can be detected. This was exemplified by investigating the system that detects physical vehicle breakins. When the anti-theft alarm is activated, the system sends messages to the lights of the vehicle to turn them on and off, so that they are flashing. An attacker does not want these lights to be activated, but since the CAN-bus is a broadcast network, messages sent by the alarm system can not be deleted (except possibly in gateways). Instead the attacker has to create new messages to turn the light off as soon as it is lit. These new messages will

be a deviation from the normal number of messages sent, and detected by the anomaly-based IDS.

3) Handling Intrusion Alerts: One crucial issue with intrusion detection is to decide what to do with an alarm that results as a consequence of a successful detection. One could think of sending the alarm to the central portal, where a security officer could take care of the alarm. However, it may not be realistic to assume that the portal should have such resources for the large amount of cars connected. Further, the car may not be continuously connected to the portal for various reasons. Thus, it seems more realistic to inform the driver of the alarms. Such an approach is proposed by Hoppe et al. [28]. By using the Human Computer Interface (HCI) various security-related events can be presented to the driver. Depending on the severity of the event, three different methods are used: (1) visual for non-critical events, (2) acoustic for critical events, and (3) haptic for severe events. They also propose an "adaptive dynamic decision model". By using the sensors of the vehicle, the environment of the vehicle can be evaluated at the time of the alert. If the currently used ways of alerting the driver is not considered to be enough, the alert-level must be increased.

4) Intrusion Prevention: So far, no Intrusion Prevention System (IPS) has been described for the vehicle settings.

In [29], Hoppe et al. discuss the problem of intrusion response and point out that an active response system might not be allowed to actively make decisions in the vehicle due to legal requirements for safety-critical systems.

D. Honeypots

A honeypot is another security mechanism that may be applied for collecting and analysing attacks against the in-vehicle network. Only one such approach has been described so far, by Verendel et al. [30]. It is suggested that the honeypot is attached to the wireless gateway in the vehicle and simulates the in-vehicle network. The data collected from the honeypot can be sent to and analysed in the portal. The purpose of this is to learn about new attacks and possibly be able to improve the next version of the system, so that it is protected against those attacks already from the beginning. One important property of the honeypot is how realistic the simulation of the target is. If the simulation is not realistic enough, the attacker may realise that he is not attacking the intended network. However, making a realistic honeypot may be very hard and Verendel et al. [30] address this by proposing three models. Another complication is that, for security and safety reasons, separate hardware should be used for the honeypot. It should also be ensured that the honeypot not detrimentally affects the real in-vehicle network.

E. Threats and Attacks

In order to classify the attacks within the automotive domain, the CERT Taxonomy by Howard and Longstaff [8] has been used, but adapted to the vehicle environment.

Also Brooks et al. [7] and Hoppe and Dittmann [19] start out from the CERT Taxonomy and adapt it for the automotive domain. Examples of new attackers added are tuners and competing manufactures; a tuner may want to manipulate the vehicle such that it gains more horse power.

V. Discussion and Summary

In the near future, or perhaps already today, the connected car will be a full-fledged node in Internet or some other IP-based network. This will give us enhanced flexibility and functionality of the services provided. At the same time we will most probably face all the threats that are channelled through Internet. Consequently, we will have to consider applying all available security mechanisms to the connected car, adapted to the specific requirements that follow from this special and highly safety-critical environment. We have found that this process has already started, but much remains to be done. The overall status for the various sub-areas that we have studied is as follows:

- *Problems in In-Vehicle Networks.* The CAN- and FlexRay-protocols still lack sufficient protection. If external communication is to be forwarded to these buses, appropriate security mechanisms need to be applied. Also, it can be noted that mechanisms implemented for safety, e.g. fault detection mechanisms in CAN, may possibly be used by an attacker to cause a security problem. Furthermore, some of the security problems are caused by poor implementations.
- *Architectural Security Features.* Two approaches use MACs to provide integrity of the messages. These approaches implement the MAC by modifying the respective protocols. Other approaches were to propose new security architectures. However, some of these approaches still have to be evaluated considering the limited resources of the in-vehicle network. Other interesting proposals are the attempt to formally verify the attestation-based security architecture as well as the concept of adding security in the vehicle through a DMS. Investigations on how such an DMS affects the in-vehicle network should be conducted.
- *Intrusion Detection Systems.* Both anomaly-based and specification-based IDSs have been suggested for the CAN-protocol. However, no approaches have been found for the other protocols. Since FlexRay also lacks appropriate security mechanisms and eventually will replace the CAN-protocol, an IDS for FlexRay should also be investigated.
- *Honeypots.* The hardest problem in implementing a honeypot is to make it separate from the real in-vehicle network and still make it as realistic as possible. If honeypots are going to be used for collecting information about attackers, further research is needed in how this can be performed in a safe and secure manner.
- *Threats and Attacks.* We note that steps have been taken in adapting the CERT Taxonomy [8] to also classify attacks towards the connected car.

As we have seen, there are already a few security features being studied for application in the vehicular environment, but many more remain to be considered. For example, Wolf et al. [18] briefly discuss the concept of a firewall, but we

know of no attempts to really introduce a firewall, where traffic is filtered at each ECU. We also note that out of the four protocols used for the in-vehicle network (CAN, LIN, MOST, and FlexRay), almost all research has been addressed to CAN. Very little work regarding the other protocols was found.

VI. Conclusion

We have surveyed the current research related to securing the connected car, with focus on the security of the in-vehicle network. Although there are some solutions proposed, most of the research has focused on identifying the security problems and only to a lesser extent towards presenting solutions. Here, much remains to be done. One of the greatest challenges in adding security to the connected car will be to adapt the security solutions to the very high safety requirements, under the constraints of very limited hardware, software and power resources.

Acknowledgements

This research was funded by the SIGYN II project (2009-01722) co-funded by VINNOVA, the Swedish Governmental Agency for Innovation Systems. It has also been supported by the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 257007.

References

[1] M. Shavit, A. Gryc, and R. Miucic, "Firmware Update Over The Air (FOTA) for Automotive Industry," in *14th Asia Pacific Automotive Engineering Conference*. Hollywood, CA, USA: SAE, 2007.

[2] S. You, M. Krage, and L. Jalics, "Overview of Remote Diagnosis and Maintenance for Automotive Systems," in *2005 SAE World Congress*, Detroit, MI, USA, 2005.

[3] 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 *Proc. of the 31st IEEE Symposium on Security and Privacy*, 2010, pp. 447–462.

[4] M. Wolf, A. Weimerskirch, and T. Wollinger, "State of the Art: Embedding Security in Vehicles," *EURASIP Journal on Embedded Systems*, vol. 2007, 2007.

[5] M. Jenkins and S. M. Mahmud, "Security Needs for the Future Intelligent Vehicles," in *2006 SAE World Congress*. Detroit, MI, USA: SAE, 2006.

[6] P. Kocher, R. Lee, G. McGraw, and A. Raghunathan, "Security as a New Dimension in Embedded System Design," in *Proc. of the 41st annual Design Automation Conference*. New York, NY, USA: ACM, 2004, pp. 753–760.

[7] R. Brooks, S. Sander, J. Deng, and J. Taiber, "Automobile Security Concerns," *Vehicular Technology Magazine, IEEE*, vol. 4, no. 2, pp. 52–64, Jun. 2009.

[8] J. D. Howard and T. A. Longstaff, "A Common Language for Computer Security Incidents," no. Sandia Report: SAND98-8667, 1998.

[9] U. E. Larson and D. K. Nilsson, "Securing Vehicles against Cyber Attacks," in *CSIIRW '08: Proc. of the 4th annual workshop on Cyber security and information intelligence research*. New York, NY, USA: ACM, 2008, pp. 30:1–30:3.

[10] D. K. Nilsson and U. E. Larson, "A Defense-in-Depth Approach to Securing the Wireless Vehicle Infrastructure," *Journal of Networks*, vol. 4, no. 7, pp. 552–564, Sep. 2009.

[11] "E-safety Vehicle Intrusion Protected Applications (EVITA)." [Online]. Available: http://www.evita-project.org/

[12] "Secure Vehicle Communication (SeVeCOM)." [Online]. Available: http://www.sevecom.org/

[13] D. K. Nilsson, U. E. Larson, and E. Jonsson, "Creating a Secure Infrastructure for Wireless Diagnostics and Software Updates in Vehicles," in *Proc. of the 27th International Conference on Computer Safety, Reliability, and Security (SAFECOMP '08)*. Newcastle upon Tyne, UK: Springer-Verlag, 2008, pp. 207–220.

[14] T. Hoppe, S. Kiltz, and J. Dittmann, "Security Threats to Automotive CAN Networks – Practical Examples and Selected Short-Term Countermeasures," in *Proc. of the 27th International Conference on Computer Safety, Reliability, and Security (SAFECOMP '08)*. Newcastle upon Tyne, UK: Springer-Verlag, 2008, pp. 235–248.

[15] D. K. Nilsson and U. E. Larson, "Simulated Attacks on CAN Buses: Vehicle Virus," in *Proc. of the 5th IASTED Int. Conference on Communication Systems and Networks*. ACTA Press, 2008, pp. 66–72.

[16] D. K. Nilsson, U. E. Larson, F. Picasso, and E. Jonsson, "A First Simulation of Attacks in the Automotive Network Communications Protocol FlexRay," in *Proc. of the International Workshop on Computational Intelligence in Security for Information Systems (CISIS'08)*. Springer Berlin / Heidelberg, 2009, vol. 53, pp. 84–91.

[17] A. Lang, J. Dittmann, S. Kiltz, and T. Hoppe, "Future Perspectives: The Car and Its IP-Address – A Potential Safety and Security Risk Assessment," in *Proc. of the 26th International Conference on Computer Safety, Reliability, and Security (SAFECOMP '07)*. Nuremberg, Germany: Springer-Verlag, 2007, pp. 40–53.

[18] M. Wolf, A. Weimerskirch, and C. Paar, "Security in Automotive Bus Systems," in *Workshop on Embedded IT-Security in Cars*, Bochum, Germany, Nov. 2004.

[19] T. Hoppe and J. Dittmann, "Sniffing/Replay Attacks on CAN Buses: A simulated attack on the electric window lift classified using an adapted CERT taxonomy," in *Proc. of the 2nd Workshop on Embedded Systems Security (WESS)*, Salzburg, Austria, 2007.

[20] T. Hoppe, S. Kiltz, and J. Dittmann, "Automotive IT-Security as a Challenge: Basic Attacks from the Black Box Perspective on the Example of Privacy Threats," in *Proc. of the 28th International Conference on Computer Safety, Reliability, and Security (SAFECOMP '09)*. Hamburg, Germany: Springer-Verlag, 2009, pp. 145–158.

[21] M. L. Chávez, C. H. Rosete, and F. R. Henríguez, "Achieving Confidentiality Security Service for CAN," in *Proc. of the 15th International Conference on Electronics, Communications and Computers, 2005. CONIELECOMP 2005.*, Feb. 2005, pp. 166–170.

[22] D. K. Nilsson, U. E. Larson, and E. Jonsson, "Efficient In-Vehicle Delayed Data Authentication Based on Compound Message Authentication Codes," in *Proc. of the 68th IEEE Vehicular Technology Conference (VTC 2008-Fall)*. IEEE, 2008, pp. 1–5.

[23] A. Groll and C. Ruland, "Secure and Authentic Communication on Existing In-Vehicle Networks," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2009, pp. 1093–1097.

[24] H. Oguma, A. Yoshioka, M. Nishikawa, R. Shigetomi, A. Otsuka, and H. Imai, "New Attestation-Based Security Architecture for In-Vehicle Communication," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*. New Orleans, LA: IEEE, 2008, pp. 1–6.

[25] C. Szilagyi and P. Koopman, "A Flexible Approach to Embedded Network Multicast Authentication," in *2nd Workshop on Embedded Systems Security (WESS)*, 2008.

[26] S. Schulze, M. Pukall, G. Saake, T. Hoppe, and J. Dittmann, "On the Need of Data Management in Automotive Systems," in *13. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS)*, vol. 144, Münster, Germany, 2009.

[27] U. E. Larson, D. K. Nilsson, and E. Jonsson, "An Approach to Specification-based Attack Detection for In-Vehicle Networks," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2008, pp. 220–225.

[28] T. Hoppe, S. Kiltz, and J. Dittmann, "Adaptive Dynamic Reaction to Automotive IT Security Incidents Using Multimedia Car Environment," in *Proc. of the 4th International Conference on Information Assurance and Security (ISIAS '08)*, Sep. 2008, pp. 295–298.

[29] ——, "Applying Intrusion Detection to Automotive IT – Early Insights and Remaining Challenges," *Journal of Information Assurance and Security*, vol. 4, no. 3, pp. 226–235, 2009.

[30] V. Verendel, D. K. Nilsson, U. E. Larson, and E. Jonsson, "An Approach to using Honeypots in In-Vehicle Networks," in *Proc. of the 68th IEEE Vehicular Technology Conference (VTC)*, 2008, pp. 1–5.

[31] L. R. Hamle and R. K. Bauer, "AINT Misbehaving – A Taxonomy of anti-intrusion techniques," in *Proc. of the 18th National Information Systems Security Conference*, 1995, pp. 163–172.

[32] G. Lee, H. Oguma, A. Yoshioka, R. Shigetomi, A. Otsuka, and H. Imai, "Formally Verifiable Features in Embedded Vehicular Security Systems," in *Vehicular Networking Conference (VNC), IEEE*, Oct. 2009, pp. 1–7.

[33] C. Szilagyi and P. Koopman, "Flexible Multicast Authentication for Time-Triggered Embedded Control Network Applications," in *Dependable Systems Networks. IEEE/IFIP Int. Conf. on*, 2009, pp. 165–174.