Experimental Evaluation of Cryptography Overhead in Automotive Safety-Critical Communication

Edilson A. Silva Junior
Centro de Informática (CIn)
Universidade Federal
de Pernambuco (UFPE)
Recife, Pernambuco, Brazil
Email: easj@cin.ufpe.br

Paulo Freitas de Araujo-Filho
Centro de Informática (CIn)
Universidade Federal
de Pernambuco (UFPE)
Recife, Pernambuco, Brazil
Email: pfaf@cin.ufpe.br

Divanilson R. Campelo
Centro de Informática (CIn)
Universidade Federal
de Pernambuco (UFPE)
Recife, Pernambuco, Brazil
Email: dcampelo@cin.ufpe.br

***Abstract*—In this paper, cryptographic schemes are applied to Ethernet-based layer-2 communication to provide authenticated encryption to safety-critical automotive control data. Confidentiality, integrity and authenticity are provided by combining AES with HMAC. Experimental results using low-cost hardware show that, despite the introduced cryptographic overhead, latency requirements are comfortably met for this type of communication.**

I. Introduction

In today's vehicles, networked functions such as connectivity, infotainment and advanced driver assistance systems (ADAS) have increasing bandwidth requirements on the internal network infrastructure of cars. Ever more signals have been transmitted via in-vehicle network systems, currently seen as a collection of single buses partly connected via gateways. This arrangement gives rise to a complex, non-homogeneous in-vehicle infrastructure that has problems of scalability, cable harness and bandwidth limitation in cars [1].

Recently, Ethernet emerged in the automotive domain as a flexible, scalable and high-bandwidth in-vehicle network solution [2]. The use of Ethernet in cars makes possible a paradigm shift to a centralized in-vehicle architecture to interconnect a growing number of heterogeneous and distributed electronic modules in a car [3].

Safety-critical automotive systems, such as the engine control, have hard real-time requirements, for which missing a deadline is not acceptable because it could compromise the safety of passengers. The authors in [4] address real-time communication with the Internet Protocol (IP) over an Ethernet-based in-vehicle backbone and state 2.5 ms as the strongest requirement concerning the end-to-end delay between two electronic modules that transmit control data. This worst case condition needed in real-time automotive communication is reinforced in [1].

The car as a node of an external vehicular network implies an exposure to security threats. In [5], the authors present a survey of security research on connected cars. In [6], the communication interfaces of a car are highlighted as entry points for cyber attacks, and security threats with their possible solutions are discussed.

A common solution utilized in automotive systems for checking data integrity is the Cyclic Redundancy Check - 32 bits (CRC32), since it does not require too much processing power [7], [8]. On the other hand, the use of cryptography for providing Authenticated Encryption (AE) is not so used in automotive systems due to the overhead introduced. As adverted in [9], it leads to additional data processing and longer messages, incurring in a significant impact on the system scalability and performance.

If Ethernet is used for safety-critical communication, it needs to support deterministic delivery of safety-critical traffic. In [10], the authors use IEEE 802.1Q and show that, by limiting the Maximum Transfer Unit (MTU) of the messages, the hard real-time requirements may be achieved without any modification to the network stack or protocols. Thus, this technique transforms a non-deterministic network into a network that deterministically meets the end-to-end delay requirement. Another strategy for bringing determinism to Ethernet is using the Audio Video Bridging (AVB)/Time Sensitive Networking (TSN) standards, however, this requires expensive specialized hardware. The introduction of authentication and optional Advanced Encryption Standard (AES) encryption is currently being developed for them, as shown in [11] and [12].

Even though there is a variety of previous works on automotive safety-critical systems and on the performance impact caused by cryptography in other domains, there is a lack of studies about these topics together. For instance, the impact caused by cryptography on performance-critical systems is studied in [13], however, not in the automotive domain. While the work in [14] proposes a steer-by-wire safety-critical system for electric cars, no security mechanism was investigated.

In this paper, we address this gap by proposing and evaluating an architecture that deploys link-layer security combining AES with Hash-based Message Authentication Code (HMAC) on an automotive safety-critical system. The proposed architecture for such a system is based on a non-deterministic isolated network for control data. Different measures, such as limiting the MTU, are applied along with a careful examination of the worst case delays to guarantee determinism. An experimental evaluation is performed using low-cost hardware supported by statistical tests on the results in order to prove that the end-to-end delay requirement is not compromised.

The remainder of this paper is described as follows. Section

978-1-5386-6355-4/18/$31.00 ©2018 IEEE

II presents the proposed architecture. Section III describes the experimental setup used. Section IV presents the results and statistically analyzes them. Finally, the main conclusions are presented in Section V.

II. Proposed Architecture

This paper proposes a secure switched Ethernet network architecture that allows the communication of automotive safety-critical control data. The architecture must guarantee that the end-to-end delay between two electronic modules that exchange control data be within 2.5 ms. Safety-critical data is isolated from lower priority messages by means of a separate network, preventing congestion of the time sensitive data [15]. Traffic shaping mechanisms and offline scheduling may be used to correctly dimension the network and avoid collisions of safety-critical data [16]. The architecture is based on a periodic time frame in which each task sends and receives messages within their reserved time slot. A synchronization protocol, such as the IEEE 1588, may be used to synchronize the clocks of the processing modules [4].

The communication is secured by applying cryptographic schemes to simultaneously guarantee confidentiality, integrity, and authenticity of safety-critical data. The network stack is configured to use only raw Ethernet frames. These frames have higher priority dedicated queues and do not have additional overhead of higher layers, what helps to minimize the impact of the introduction of cryptography to the data. As shown in Figure 1, the sequence of numbers points out the dedicated path taken by a given packet, which does not go through the TCP/IP stack – raw packets are serviced to completion before any IP packet. The performance of the frame transmission is further enhanced by using the no-copy configuration for raw Ethernet frames. This prevents copies of the data buffer, avoiding additional overhead due to memory allocation for copying frames [17].

To establish a secure channel, AE is applied to the link layer using the Encrypt-then-MAC approach. This strategy has advantages over other common methods, such as MAC-then-Encrypt and MAC-and-Encrypt [18]. Confidentiality is provided by the AES algorithm by encrypting the frame payload. To provide integrity and authenticity, a Message Authentication Code (MAC) is calculated by the HMAC algorithm, using as input the frame header plus the already encrypted payload. This MAC is then concatenated with the encrypted payload. AES and HMAC are the most popular algorithms for AE [19].

The secured communication platform can be applied to any automotive safety-critical system. A possible use case could be a steer-by-wire system for an electric car [14].

III. Experimental Setup

In order to evaluate the proposed architecture, an experimental prototype has been built. For simplicity, the prototype consists of an Ethernet switch and two nodes, as depicted in Fig. 2. The platform is designed to support several nodes depending on the size of the payload used for a given task and

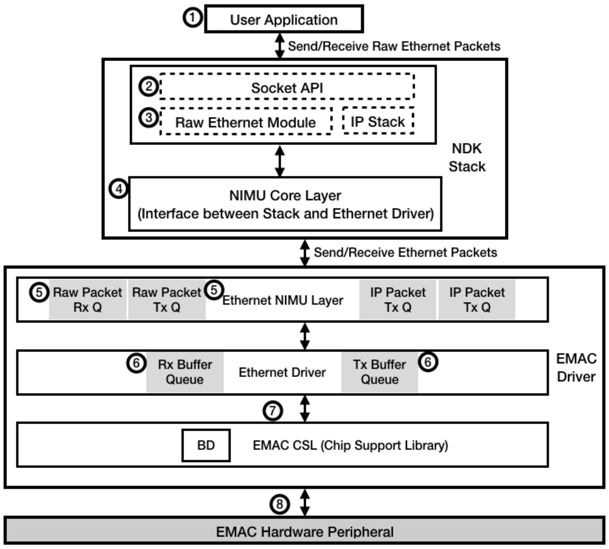

Fig. 1. Network stack of the Network Development Kit (NDK) used. The sequence of numbers indicates the dedicated path for raw Ethernet data [17].

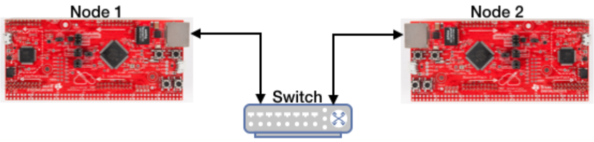

Fig. 2. Experimental setup.

its respective Time Division Multiple Access (TDMA)-based slot. All slots must be considered during the offline scheduling phase of the isolated network. In the prototype, one of the nodes represents the main processing unit of a car subsystem and the other one the processing unit responsible for some local activity, such as controlling the wheels of an axle-less electric car [14].

Tiva C Series TM4C129E boards [20], from Texas Instruments (TI), are used as network nodes along with an off-the-shelf household switch. The boards are low-cost solutions with embedded dedicated hardware modules, which are used in all of our experiments involving cryptography or CRC32. A simple switch is sufficient to assess the impact of cryptography on the end-to-end delay of control data. In addition, a common run-time software is utilized on all nodes in the form of a Real-Time Operating System (RTOS).

A. Scenarios

The performance of the AES algorithm significantly varies when security keys of different sizes are used. Similarly, the performance of HMAC can be different depending on the underlying algorithm utilized. The dedicated cryptography module supports AES security key sizes of 128, 192 and 256 bits, and the underlying HMAC algorithms SHA-MD5, SHA-1, SHA-224 and SHA-256. All these key sizes and algorithms are evaluated in this paper regarding their performance. The

TABLE I
POSSIBLE COMBINATIONS USING CRYPTOGRAPHY

Payload size	AES	HMAC
64 bytes	128 bits	SHA-MD5 (16 bytes)
128 bytes	192 bits	SHA-1 (20 bytes)
736 bytes	256 bits	SHA-224 (28 bytes)
1440 bytes		SHA-256 (32 bytes)

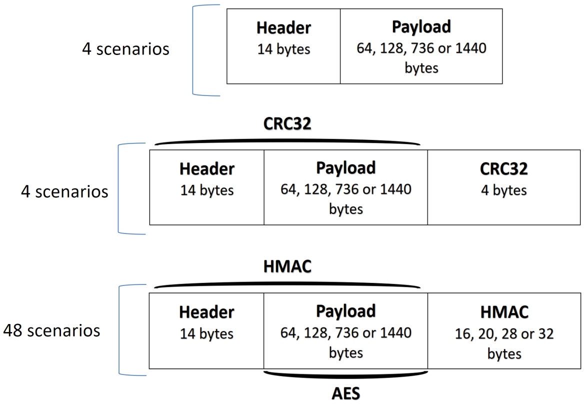

Fig. 3. Possible scenarios for the experiments.

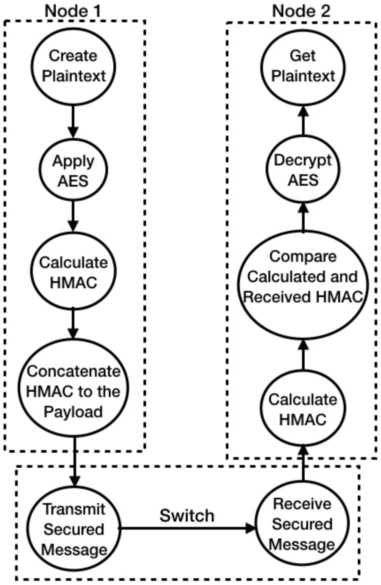

Fig. 4. Experiment flowchart.

objective is to provide a guideline when using different complexity schemes that result in different end-to-end delays.

A common practice to guarantee the deadlines for safety-critical data in non-deterministic networks is to limit the MTU of the messages [10]. As a result, the impact of the message size is also investigated in this paper. Four payload sizes are used: 64, 128, 736 and 1440 bytes. While 64 and 128 bytes are reasonable payload sizes for safety-critical data, 736 and 1440 bytes are chosen to roughly represent 50% and 100% of the maximum payload size in Ethernet frames, respectively. A margin of at least 32 bytes is added to the payload sizes due to HMAC concatenation. Table I groups together the considered payload sizes, the AES key sizes and the four underlying algorithms for HMAC, which lead to 48 possible combinations using cryptography.

Figure 3 describes the three frame formats used in the experiment: a plain frame, a frame that uses CRC32 and a frame that uses cryptography. Four scenarios with plain frames offer no security and differ among themselves only by the payload size. Four scenarios with CRC32 only verify for data integrity and also differ among themselves by the payload size. The remaining 48 scenarios offer AE by using AES and HMAC and differ among themselves not only by the payload size, but also by the AES key size and the underlying HMAC algorithm.

The experiment using cryptography is divided in three parts, as depicted in the flowchart of Fig. 4. The first part consists of Node 1 creating a message, applying the AES algorithm, and calculating and concatenating the corresponding HMAC to the resulting frame. The second part corresponds to the elapsed time from the instant Node 1 starts transmitting the secured message until it is completely received by Node 2. The third part consists of the inverse process of the first part, i.e., calculating HMAC to check it against the received HMAC in the frame and applying the AES in the reverse direction to decrypt the message. For each of the described 56 scenarios, the experiment is reproduced 3600 times. The end-to-end delay is measured using TI's instrumentation software, recommended for evaluating real-time applications without impacting their performance.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The main goals of the experiments are to verify whether each scenario that uses cryptography satisfies the maximum end-to-end delay of 2.5 ms and to compare their delay performances. The scenarios that use plain frames and CRC32 are baselines for the comparisons.

An initial analysis consists of experimentally measuring the end-to-end delay for each one of the 56 scenarios. The lowest, mean and highest measured values in each scenario for payloads with size 64, 128, 736 and 1440 bytes are illustrated, respectively, in Figures 5, 6, 7 and 8. From the figures, one can observe that the end-to-end delays of all scenarios are within the 2.5 ms limit and that the payload size has a great contribution to the delay, as expected. As discussed before, limiting the MTU is a common practice to guarantee the deadlines for safety-critical data in non-deterministic networks. However, it is known that this procedure decreases the frame efficiency. In any case, control data from safety-critical applications are usually small in size.

Another observation is the large distance between the highest and the mean values of the end-to-end delays. This is because some outliers with a considerably longer end-to-end delay are observed for each scenario. These outliers are a result of the way the RTOS allocates and frees memory during the transmission of the frames. While the replications for each

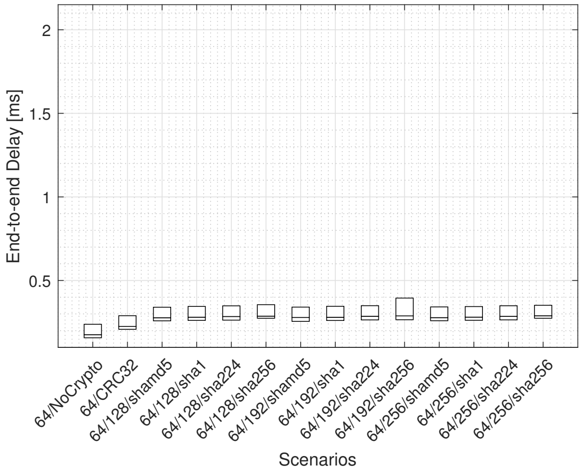

Fig. 5. End-to-end delay for 64-byte payloads.

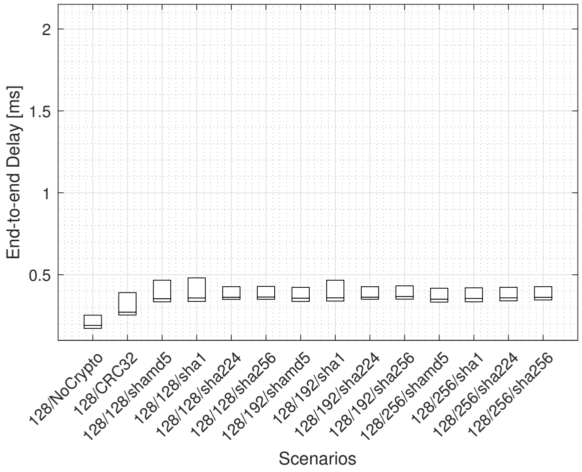

Fig. 6. End-to-end delay for 128-byte payloads.

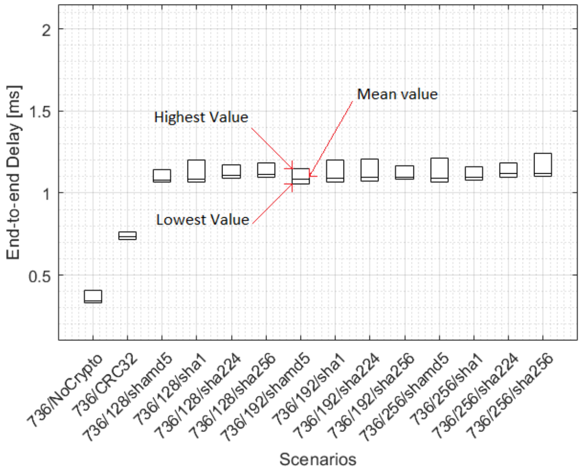

Fig. 7. End-to-end delay for 736-byte payloads.

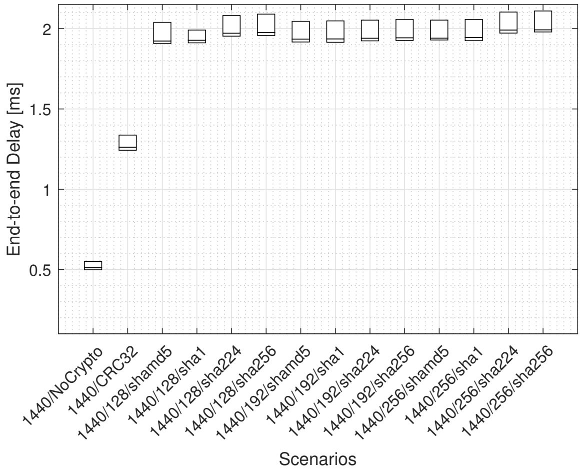

Fig. 8. End-to-end delay for 1440-byte payloads.

scenario are executed, even when the no-copy variation is used, the RTOS intrinsically performs a memory deallocation of the buffers after a frame is sent. This leads to a peak in the resulting delay, which is more frequent if the size of the payload is increased. For safety-critical applications, this extra time must be considered since it is not affordable to miss any deadline.

In addition to the measurements complying with the limit value, a statistical test is performed for ensuring that the obtained results are concentrated below and distant from this limit. Since the results obtained from the experiments do not follow a normal distribution, a non-parametric statistical test is performed on the results. In this paper, a Wilcoxon Unilateral statistical test is used.

This hypothesis test states with 99% confidence that the median of the experiment is statistically lower than the threshold value tested for each scenario. The threshold values tested are chosen to be significantly lower than the 2.5 ms limit. For instance, for the 256/SHA256 scenario, the smallest threshold values for the tested payload sizes are shown in Table II.

In order to compare the scenarios, the non-parametric statistical Friedman test is conducted. The Friedman test is performed by fixing two of the characteristics of the schemes and varying the third one. The goal is to rank the schemes with respect to the end-to-end delay associated to them. The null hypothesis, which states that the scenarios are equal, was rejected. Due to space limitations, only some of the results

TABLE II
WILCOXON UNILATERAL STATISTICAL TEST THRESHOLD FOR THE EXPERIMENT MEDIAN

Payload Size	AES Key Size	HMAC	Median Threshold
64 bytes	256 bytes	SHA256	0.289 ms
128 bytes	256 bytes	SHA256	0.363 ms
736 bytes	256 bytes	SHA256	1.121 ms
1440 bytes	256 bytes	SHA256	1.993 ms

TABLE III
FRIEDMAN TEST, 128-BYTE PAYLOAD AND THE SAME UNDERLYING ALGORITHM FOR HMAC; THE AES KEY SIZE IS VARIED

Payload Size	Cryptography Scheme	Ranking
128 bytes	No Crypto	3
128 bytes	CRC32	8.0233
128 bytes	256 bytes AES Key / SHA256	14.0585
128 bytes	128 bytes AES Key / SHA256	17.5775
128 bytes	192 bytes AES Key / SHA256	22.3407

TABLE IV
FRIEDMAN TEST, 128-BYTE PAYLOAD AND THE SAME AES KEY; THE UNDERLYING ALGORITHM FOR HMAC IS VARIED

Payload Size	Cryptography Scheme	Ranking
128 bytes	No Crypto	3.5
128 bytes	CRC32	9.5378
128 bytes	256 bytes AES Key / SHAMD5	16.281
128 bytes	256 bytes AES Key / SHA1	21.4631
128 bytes	256 bytes AES Key / SHA224	27.4811
128 bytes	256 bytes AES Key / SHA256	32.7371

of the application of the Friedman test are presented in this paper.

Table III presents the results of the Friedman test for the comparison of schemes with 128-payload size, the same underlying algorithm for HMAC and different values for the AES key size. As expected, the ranking obtained from the test shows that the "No Crypto" and CRC32 cases present the best end-to-end delays in this order. On the other hand, the results do not support the intuitive assumption that bigger keys would incur in longer delays.

Table IV shows the results of the Friedman test for the comparison of schemes with 128-byte payload size, the same AES key size and different underlying HMAC algorithms. Here however, as it was to be expected, the "No Crypto" and CRC32 cases are the ones with the best end-to-end delays and, the higher the HMAC algorithm complexity, the longer the delay.

V. CONCLUSION

This work evaluates the suitability of authenticated encryption in automotive safety-critical applications regarding end-to-end delay constraints. It proposes an architecture based on link-layer security in automotive and deploys it with a low-cost experimental prototype. A predictable behavior is achieved within an isolated network for secured safety-critical control data by applying the measures described in the paper.

Although the proposed architecture does not aim to provide the best security solution, it intends to offer a performance guideline for engineers and researchers when exploring security aspects in safety-critical applications for automotive. Forty-eight different combinations of cryptography schemes are successfully evaluated and all of them comfortably achieved the end-to-end delay timing requirement.

Future works include incorporating synchronization protocols, such as Precision Time Protocol (PTP) and Generalized Precision Time Protocol (gPTP), into the architecture. The results from this paper indicate that the use of synchronization protocols along with AE might compromise the end-to-end delay for bigger payload sizes. Even though safety-critical applications use smaller payload sizes, further investigation is required.

ACKNOWLEDGMENT

This work was supported in part by FACEPE and CNPq.

REFERENCES

[1] S. Tuohy, M. Glavin, C. Hughes, E. Jones, M. Trivedi, and L. Kilmartin, "Intra-vehicle networks: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 534–545, 2015.

[2] L. L. Bello, "The case for Ethernet in automotive communications," *ACM SIGBED Review*, vol. 8, no. 4, pp. 7–15, 2011.

[3] H. Staehle, L. Mercep, A. Knoll, and G. Spiegelberg, "Towards the deployment of a centralized ICT architecture in the automotive domain," in *2013 2nd Mediterranean Conference on Embedded Computing (MECO)*. IEEE, 2013, pp. 66–69.

[4] R. Steffen, R. Bogenberger, J. Hillebrand, W. Hintermaier, A. Winckler, and M. Rahmani, "Design and realization of an IP-based in-car network architecture," *The First Annual International Symposium on Vehicular Computing Systems, Dublin, Ireland*, pp. 1–7, 2008.

[5] P. Kleberger, T. Olovsson, and E. Jonsson, "Security aspects of the in-vehicle network in the connected car," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 528–533.

[6] I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaâniche, and Y. Laarouchi, "Survey on security threats and protection mechanisms in embedded automotive networks," in *Dependable Systems and Networks Workshop (DSN-W), 2013 43rd Annual IEEE/IFIP Conference on*. IEEE, 2013, pp. 1–12.

[7] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, 2015.

[8] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security–a survey," *arXiv preprint arXiv:1701.04525*, 2017.

[9] S. Chakraborty, M. A. Al Faruque, W. Chang, D. Goswami, M. Wolf, and Q. Zhu, "Automotive cyber–physical systems: A tutorial introduction," *IEEE Design & Test*, vol. 33, no. 4, pp. 92–108, 2016.

[10] Y. Lee and K. Park, "Meeting the real-time constraints with standard Ethernet in an in-vehicle network," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 1313–1318.

[11] J. Koftinoff, "Avb/tsn developer faq," 2016, [Online; accessed 5-February-2017]. [Online]. Available: https://avb.statusbar.com/page/developer-faq/

[12] J. Holle and T. Lothspeich, "Security concepts for Ethernet based e/e-architectures," in *IEEE-SA Ethernet I& IP @ Automotive Technology Day 2016, Paris, France*. IEEE, 2016.

[13] W. Freeman and E. Miller, "An experimental analysis of cryptographic overhead in performance-critical systems," in *Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 1999. Proceedings. 7th International Symposium on*. IEEE, 1999, pp. 348–357.

[14] C. Buckl, A. Camek, G. Kainz, C. Simon, L. Mercep, H. Stähle, and A. Knoll, "The software car: Building ICT architectures for future electric vehicles," in *Electric Vehicle Conference (IEVC), 2012 IEEE International*. IEEE, 2012, pp. 1–8.

[15] A. G. Camek, C. Buckl, and A. Knoll, "Future cars: necessity for an adaptive and distributed multiple independent levels of security architecture," in *Proceedings of the 2nd ACM international conference on High confidence networked systems*. ACM, 2013, pp. 17–24.

[16] M. Rahmani, K. Tappayuthpijarn, B. Krebs, E. Steinbach, and R. Bogenberger, "Traffic shaping for resource-efficient in-vehicle communication," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 4, pp. 414–428, 2009.

[17] *TI Network Developer's Kit (NDK) v2.25 API Reference Guide*, Texas Instruments, Jan. 2016, rev. J.

[18] M. Bellare and C. Namprempre, "Authenticated encryption: Relations among notions and analysis of the generic composition paradigm," *Advances in CryptologyASIACRYPT 2000*, pp. 531–545, 2000.

[19] W. Stallings, *Cryptography and network security: principles and practices*. Pearson Education India, 2006.

[20] *Tiva TM4C129ENCPDT Microcontroller DATA SHEET*, Texas Instruments, Jun. 2014, rev. B.