Securing Vehicular On-Board IT Systems: The EVITA Project

Dr.-Ing. **Olaf Henniger**, Fraunhofer Institute SIT, Darmstadt
Dr. **Alastair Ruddle**, MIRA Ltd, United Kingdom
Dipl.-Inf. **Hervé Seudié**, Robert Bosch GmbH, Stuttgart
Dr.-Ing. **Benjamin Weyl**, BMW Group Forschung und Technik, München
Dr.-Ing. **Marko Wolf**, escrypt GmbH, München
Dr.-Ing. **Thomas Wollinger**, escrypt GmbH, Bochum

Abstract

Secure and trustworthy automotive on-board IT systems are essential for the success of intelligent transport systems based on vehicle-to-vehicle and vehicle-to-infrastructure communication. The EVITA project therefore aims to design, verify, and prototype suitable architectures for secure automotive on-board networks, thus complementing other projects that focus on protecting communications between vehicles and between vehicles and infrastructure entities. This paper outlines the approach followed in the EVITA project – from the analysis of security requirements through the hardware and software design up to the planned prototype – and summarizes first results.

1 Introduction and Objectives

Today the need for vehicular IT security measures is undisputed. Vehicle theft, circumvention of restrictions on vehicular functionality (e.g., feature activation, software updates), manipulation of vehicular components with financial, legal, or warranty implications (e.g., toll devices, digital tachograph, chip tuning) or theft of intellectual property (e.g., counterfeits, industrial espionage) are amongst the most prominent threats demonstrating a strong need for vehicular IT security measures. While the integration of wireless communication technologies inspires new applications, for instance, safety applications based on vehicle-to-infrastructure (V2I) or vehicle-to-vehicle (V2V) communication such as traffic light preemption, or emergency break notification, new security requirements need to be considered in order to prevent attacks on such V2X systems.

From a security perspective, most vehicular on-board IT systems operate in a “hostile” environment, similar to pay-TV set-top boxes, where potential attackers have virtually unlimited time and an unlimited number of trials and, most importantly, full physical access to manipulate their own devices. Fortunately, many capable vehicular IT security measures already exist, such as authentication schemes, encrypted memories, or encrypted communications, which cover many security requirements already. V2X applications require new security measures, such as the protection against illegally forced malfunctioning of safety critical in-vehicular components, either via physical access or by means of external communication interfaces. We need an appropriate hardware security architecture that protects also against physical in-vehicle attacks. The reason behind this is that dependable security measures must at least be able to trust their underlying hardware; in particular, in order to rule out that vehicular attackers could get access to in-vehicular security mechanisms (e.g., to read-out secret keys, exchange authentication credentials, or simply deactivate security functionalities). The hardware security architecture should provide security mechanisms for the secure generation, secure storage, and secure processing of security-critical material (e.g., secret keys), whilst being additionally shielded from potential malicious intrusions with the help of tamper protection measures, which require significant technical and financial efforts to break them.

An approach to provide a cost-effective hardware security architecture will be the outcome of the EVITA project (*E-safety Vehicle Intrusion proTected Applications*) of the Seventh European Framework Program, which aims to design, verify, and prototype a security architecture for vehicular on-board networks, where all security-relevant components and sensitive information are protected against tampering and malicious manipulations [1]. Thus, EVITA implements the security technology for vehicular communication endpoints, enabling the security of safety applications, but also most other V2X communication applications (e.g., vehicular comfort, or business applications).

This publication provides an insight into the projected approach. It introduces relevant e-safety use cases and briefly presents the corresponding threat analysis and security requirements engineering process. The publication further outlines the projected realization of the underlying security hardware architecture. We further provide an outlook on the integration of the hardware architecture with software components.

2 Example Use Cases and Stakeholders

Within EVITA the following general use case categories are considered in order to cover most of the several objectives specifically related to the security of an on-board IT system [2]:

- communication between cars (e.g. local danger warning),
- communication between car and infrastructure (e.g. eCall),
- integration of mobile devices (e.g., CE devices or smart-phones),
- aftermarket applications (e.g. feature activation), and
- workshop and diagnosis processes (e.g. software updates or remote diagnosis).

As the SeVeCom project has specifically focused on the communication between vehicles and infrastructure entities, EVITA could select significant use cases from [3] in order to focus on the analysis of threats and infer security requirements related to the critical assets of the vehicular on-board network.

The main assets of a vehicular on-board IT system that may become targets of attacks are on-board electronic components such as ECUs, sensors, actuators, and the communication links between these components (e.g. the CAN bus) as well as the communication links at application-level within ECUs. In consideration of the use cases, the interests of the stakeholders involved comprise at least the following:

- *Vehicle occupants*: safe and efficient driving, valid financial transactions, personal privacy, protection of personal data;
- *Other road users*: safety and efficient transport;
- *Vehicle manufacturers and suppliers*: successful and affordable satisfaction of customer expectations, safe, efficient and privacy preserving deployment and operation of applications, protection of intellectual properties and know-how;
- *ITS system providers*: safe and efficient operation of systems, valid financial transactions, protection of user data;
- *Civil authorities*: safe and efficient transportation networks, reliable financial transactions, data protection.

With the identified use cases, stakeholders and protectable assets, a threat, risk and security requirements analysis has been performed accordingly.

3 Threat, Risk and Security Requirements Analysis

The aims of the EVITA threat, risk and security requirements analysis was to derive, justify, and prioritize security requirements and security related safety requirements for automotive on-board networks. The security engineering process developed for this purpose is described in [4][5][6]. The process includes the identification of potential threats for each use case harming a dedicated asset and stakeholders' security objectives. Based on these asset attacks the level of risk posed by the attack has been assessed. The risk is a function of the possible severity and the required attack potential. For the purpose of EVITA, severity is a function of the impact on safety, privacy infringements, financial loss, and operational functionality not directly related to safety. The attack potential is a measure of the minimum effort to be expended in successfully mounting an attack and has been based on the Common Criteria approach [9]. The risk level is then determined from the severity and the attack probability (i.e. the inverse of attack potential) of a specific attack. A brief overview of the main elements of this process is described in the following subsections.

3.1 Threat Analysis

At the highest level, the security objectives are:

- to ensure the safety of the vehicle occupants and other road users,
- to maintain the intended operational performance of all vehicle and ITS functions,
- to protect the privacy of vehicle drivers, and the intellectual property of vehicle manufacturers and their suppliers,
- to prevent fraudulent commercial transactions and theft of vehicles.

Potential threat agents include dishonest drivers, hackers, criminals, and terrorists, dishonest organizations and rogue states.

The starting points for the security requirements analysis were the EVITA use cases [1] and the generic network architecture based on the EASIS project [7]. Potential threats were identified and documented using an attack tree approach [8]. The attack trees were also used to work back from the anticipated motivations of various classes of attacker that were considered (i.e., organizational, professional and casual) through to possible combinations of particular attacks on specific system assets that could lead to the achievement of the attacker's objectives. The initial attack trees were subsequently re-structured in order to better support risk analysis and to identify common attack patterns.

3.2 Risk Analysis

Relative risks were associated with the threats by assessing relative severity at the higher levels of the attack tree and working up probabilities from the terminal nodes. The latter are based on “attack potential” estimates derived using an approach similar to the one described in the Common Criteria [9], with relative attack probability taken to be the inverse of attack potential. Relative risk was then attributed based on combinations of severity and probability: high probability attacks with severe outcomes are considered to be high risk, while low probability attacks with minor outcomes are considered to be low risk.

The assessment of relative severity was based on the severity classification used in vehicle safety engineering [10], augmented to reflect the potential for multiple vehicles to be involved and the fact that aspects other than safety such as financial, privacy, and operational aspects may be compromised by the attack. Operational aspects include illegal interference with vehicle systems or functions that do not directly impact on functional safety, but may cause user annoyance. This approach results in a “severity vector” with four components that may have different ratings. The components translate to different relative risk levels. For example, it is possible that an attack could have little or no impact on safety, but still present significant risks in terms of compromised driver privacy or loss of reputation for vehicle manufacturers. Consequently, the relative risks are also represented (in general) by 4-component vectors.

The results of the risk analysis were summarized in terms of the number of instances of particular risk levels found for each of the attacks that could be envisaged against the various system assets. This gives an indication of the relative importance of protecting against specific asset attacks.

3.3 Identification of Security Requirements

Investigation of the security requirements was based on a number of key security properties. These included confidentiality, privacy, non-repudiation, access control, availability, integrity, authentication of origin and freshness.

Security requirements were identified using two different but complementary viewpoints:

- Abstract functional path – based on a purely functional representation of the use cases, providing security requirements by class (confidentiality, authenticity etc.),
- Detailed functional path and mapping – based on mapping a functional representation of the use cases to an architecture, providing both functional and architectural (availability, timing) requirements by use case.

The abstract functional path approach provides a very compact description of V2X communications and a systematic approach to the identification of their associated security requirements. The detailed functional path and mapping approach allows aspects such as availability and timing, and dependencies between requirements, to be considered. Merging the results of these two viewpoints should ensure that the security requirements are sufficiently comprehensive to support subsequent design activities.

There are inevitably cost implications in meeting additional requirements. The security requirements identified in EVITA are therefore mapped to the asset attack summary resulting from the threat and risk analysis activity in order to prioritize the security requirements. This provides a mechanism for ensuring that finite development budgets can be targeted to protect the vehicle systems against the most significant of the anticipated security threats.

3.4 Summary of EVITA Security Requirements

With the threat and risk analysis performed within EVITA, the following key security requirements have been identified:

(1) *Integrity of hardware security module*: Tampering with hardware security modules (i.e., breaching the cryptographic boundary) needs to be prevented or must at least be detectable (e.g., by random inspections).

(2) *Integrity and authenticity of in-vehicle software and data*: Unauthorized alteration of any in-vehicle software and (locally stored) data involved in e-safety applications must be infeasible or must at least be detectable by the platform itself by verifying that a software or data was truly created by its claimed authors.

(3) *Integrity and authenticity of in-vehicular communication*: Unauthorized modification of messages and data which is sent within the in-vehicular network can be detected by the receiver, while verifying that a message was truly and lastly created by its claimed author.

(4) *Confidentiality of in-vehicular communication and data*: Unauthorized disclosure of confidential messages[1] and data sent or stored in the vehicle must be infeasible. Confidential messages and data must be intelligible only to their authorized recipients.

(5) *Proof of platform integrity and authenticity to other (remote) entities*: An entity must be capable to prove the integrity and authenticity of its platform hardware and software configuration to other entities.

---

[1] Which messages are required to be confidential depends on the use cases.

(6) *Access Control to in-vehicle data and resources*: In order to enable availability and well-defined access to all data and resources required by e-safety applications, the requestor of a functionality or data must be authenticated and authorized in order to gain permission before being able to use e-safety-relevant resources or access data.

The EVITA security requirements have been formally described associating the requirements to authenticity and confidentiality, which can then be mapped to a formal security modeling framework in order to evaluate the requirements on a specified architecture. For a detailed description of security requirements engineering process please refer to [4].

4 Architectural Design

The following section introduces the design of the EVITA hardware security architecture followed by the integration of the hardware security architecture with software components.

4.1 Hardware Design

The security requirement engineering process allowed us to derive the security goals related to the considered use case categories. The architecture presented in this section in turn is designed to meet these goals. The architecture applies dedicated hardware security modules (HSM), which are hardware devices that enable the secure and efficient implementation of cryptographic operations and the secure operation of dedicated security mechanisms.

Dedicated hardware for security is a key issue in the automotive field, since their use cannot be dissociated from the implicated costs. The challenge is to design the most cost-effective architecture while not compromising the security objectives. The approach was to identify which (functional and security) requirements each type of HSM for a particular component (e.g., sensor, ECU) has to fulfill. The goal was not to have a full HSM for each in-vehicle IT component. Indeed, we first sought to identify how many different types of HSMs are needed. Secondly, we reduced the properties of each HSM type to meet the requirements of the target component type. We have identified three different classes of HSMs:

- the “EVITA HSM Full Version” as hardware extension to the ECU specifically responsible for V2X applications,
- the “EVITA HSM Medium Version” as hardware extension to the ECU connected to the in-vehicle domain controls (e.g., power train control) and,
- the “EVITA HSM Light Version” for security-critical sensors and actuators.

These three different classes of HSMs allow meeting (i) the different cost constraints, (ii) the different security requirements, and (iii) the different (security) functional requirements.

Figure 1 shows an exemplary vehicular deployment architecture where all three EVITA HSM classes are used to protect security-critical components according to their individual needs and capabilities in order to provide an efficient, holistic in-vehicle security architecture.

The internal structure of our three proposed EVITA HSM types is described in more detail in the proceeding subsections.

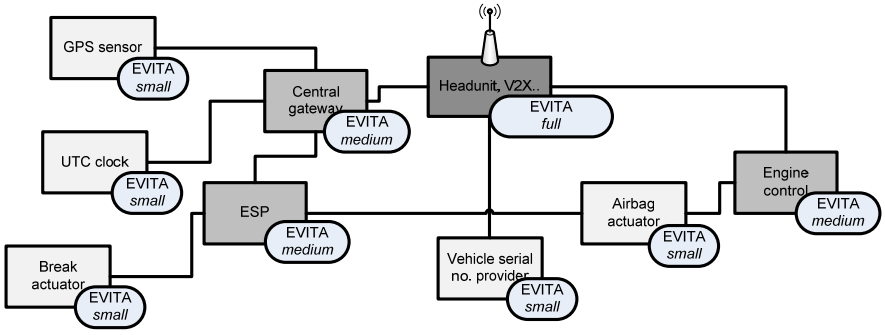

Figure 1 EVITA deployment architecture

4.1.1 EVITA Hardware Security Module Full Version

In order to satisfy the performance requirements for signing and verifying messages for V2X communications, a very efficient asymmetric cryptographic engine is required. In this case, the EVITA HSM Full Version is applied, which provides the maximum level of functionality, security, and performance. It further aims to provide a security lifetime of at least 20 years, which means ECRYPT II Level 7 "Long-term protection" [11] and/or NIST 2030+ [12].

Figure 2 depicts the architecture of the EVITA HSM Full Version, which generally consists of two parts, namely, (i) the cryptographic building block that realizes all cryptography hardware

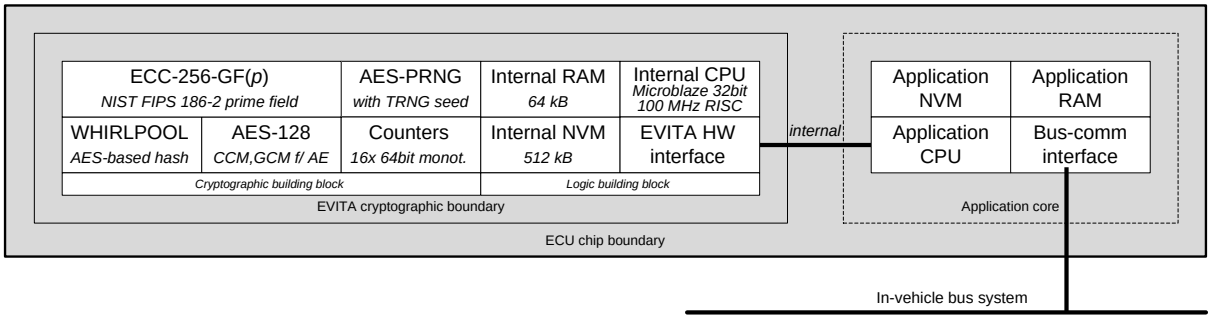

Figure 2 *Full* version of the EVITA hardware security module

operations and (ii) the logic building block that connects the EVITA hardware with the ECU application core and that (i.e., the internal logic building block) optionally may execute also some cryptographic operations in software. The cryptographic building block of the EVITA HSM Full Version is composed of the following components:

- *ECC-256-GF(p)* is a high-performance asymmetric cryptographic engine based on a high-speed 256-bit elliptic curve arithmetic using NIST approved prime field parameters [13]. It can generate/verify around 200 signatures per second while requiring about 2,000 slices FPGA hardware size.
- *WHIRLPOOL* is an AES-based hash function as proposed by NIST. It has a throughput of about 1,000 Mbit/s while requiring about 3,000 slices FPGA hardware size [14].
- *AES-128* is a symmetric block encryption/decryption engine using the official NIST advanced encryption standard. It supports not only standard block encryption modes of operation such as ECB and CBC, but also advanced encryption as used, for instance, in authenticated encryptions schemes such as GCM or CCM. Our proposed prototypical implementation has a throughput of about 1,000 Mbit/s while requiring about 1,000 slices FPGA hardware size, however, almost arbitrary optimizations for size or performance are possible [15].
- *AES-PRNG* is a pseudo random number generator, which is nonetheless seeded with a true random seed from a true internal physical random source. It is again based on an internal AES engine according to BSI-AIS20-E.4 [16]. Thus, throughput and size are defined by the underlying block cipher implementation. By using our AES implementation, we achieve throughputs of several hundred Mbit/s and about some additional hundred slices for the PRNG control logic.
- *COUNTER* is a 64-bit monotonic counter function block that serves as a simple secure clock alternative. It provides at least 16 counters together with corresponding access control that can be increased only. Each counter can be increased at least with 1 Hz while requiring about 100 slices FPGA hardware size.

Finally, the EVITA HSM Full Version uses an own independent internal CPU that can directly access its internal RAM and non-volatile memory to prevent any malicious interferences from the application CPU and the application software. The application CPU and its applications can access the EVITA security hardware only using the secure EVITA hardware interface that enforces a well-defined access (e.g., preventing the read-out of secret keys). Hence, for the internal processing, the EVITA HSM Full Version uses the following logic building blocks:

- *CPU* is an internal 32-bit microprocessor – prototypically realized by a *Microblaze* FPGA soft-core [17] – that can handle all logics and non-time-critical cryptographic functionality (e.g., high-level operations). However, it cannot be used to execute high-performance consuming cryptographic algorithms in an efficient manner. It operates at about 100 MHz while requiring about 2,000 slices FPGA hardware size.
- *RAM* is a small volatile memory to store for instance intermediate values and variables. It has a capacity of at least 64 kByte and will be realized using available block RAM memories (576 kB) and available SDRAM of the evaluation board (~64 MB)
- *NVM* is a small non-volatile memory to store for instance internal keys and security certificates. It has a capacity of about 512 kByte and will be realized using available external flash memories of the evaluation board (~16 MB).
- *HW-API* is the secure hardware interface that enforces a well-defined access to the hardware security functionality for the application software. It provides (with the help of the internal CPU) message pre- and post-processing (e.g., payload extractions, padding, etc.), message/session management and message/session control.

Table 1 gives a final summary of the size and performance estimations for all FPGA building blocks used in the full version EVITA HSM.

Building block	**FPGA size (slices) estimation**	**Performance estimation**
ECC-256-GF(p)	2,000	200 sig/s
WHIRLPOOL	3,000	1 Gbit/s
AES-128	1,000	1 Gbit/s
PRNG	200*)	1 Gbit/s
COUNTER	100	16x 64bit
CPU	2,000	32bit-RISC, 100MHz, 100 DMIPS
RAM	N/A (use block RAM)	64 kB
NVM	N/A (external)	512 kB
CONTROL / IF	1,000 – 2,000	N/A
Total	~ 10,000	

Table 1: Size and performance estimations for EVITA hardware implementation
*) Additional slices based on the assumption that the underlying block cipher is already available

4.1.2 EVITA Hardware Security Module Medium Version

The EVITA HSM Medium Version was designed to suit both the stringent security requirements and the ability to cost-effectively deploy respective security mechanisms on powerful ECUs such as those used as gateways or for engine control. **Fehler! Verweisquelle konnte**

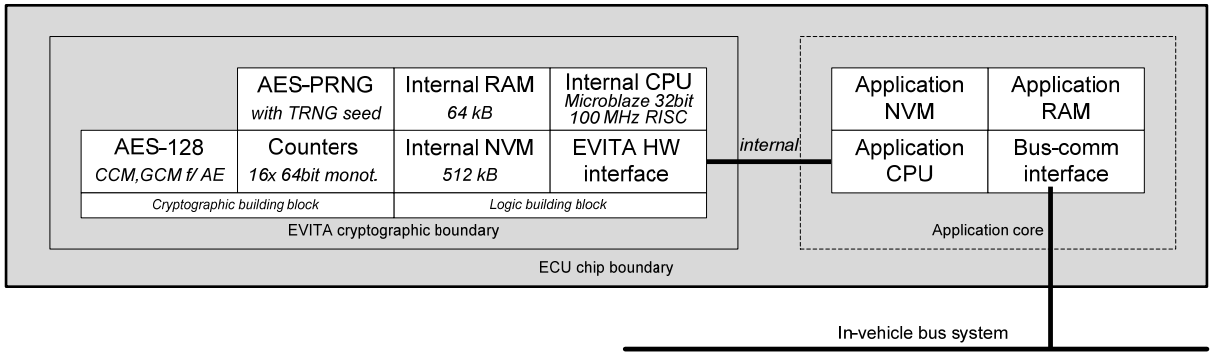

Figure 3 *Medium* version of the EVITA security module

nicht gefunden werden. depicts the EVITA HSM Medium Version, which is virtually identical to the full version except in that it has no hardware ECC engine and no hardware hash engine. Hence, the EVITA HSM Medium Version can execute very fast symmetric cryptography in hardware, but rather slow asymmetric cryptography in software (e.g., for secure key establishment or occasional digital signature processing). Note that (similarly to the full version) all security credentials are permanently protected since they are permanently out of reach of the application CPU.

4.1.3 EVITA Hardware Security Module Light Version

Figure 4 depicts the EVITA HSM Light Version designed to integrate and protect ECUs, sensors and actuators that provide or process security critical information. At this level, the EVITA security protection scheme is reduced to a single very specially-optimized symmetric AES hardware accelerator [18], while all security credentials are handled by the application processor. Even though this approach cannot provide any hardware-based security, it enables sensors and actuators to process and generate protected information in an efficient manner using their application core together with the EVITA hardware AES accelerator.

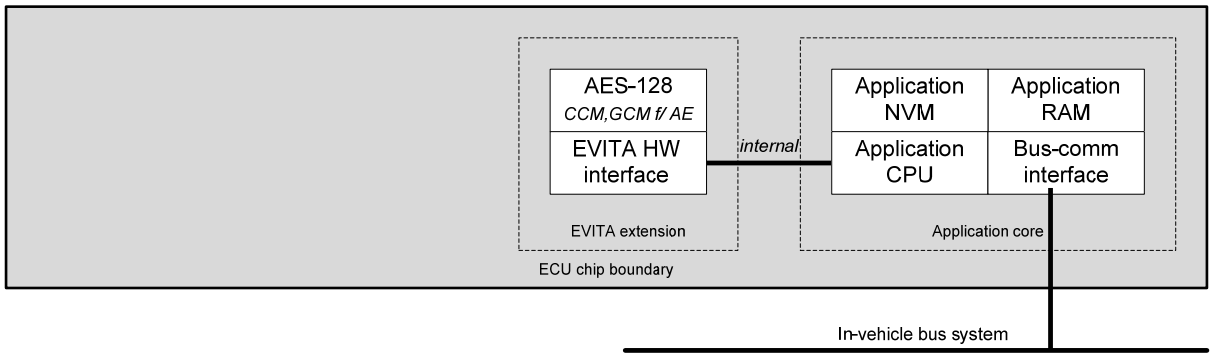

Figure 4 *Light* version of the EVITA security module

4.2 Software Design

Since the project is currently on the first third of its duration, the software design now is rather a basic concept than a final design. The basic idea is to integrate the EVITA hardware modules in the Automotive Open System Architecture (AUTOSAR) that “will serve as a platform for most future vehicle applications” [19].

The AUTOSAR software architecture is basically composed of four layers: application layer, services layer, ECU Abstraction layer and Microcontroller Abstraction layer. In order to be compatible to AUTOSAR, EVITA will define adapted modifications for each layer. The security applications will logically be added to the application layer and will access the cryptographic routines of the services layer through an API, which must be well-defined. An extended security library will be implemented in the services layer. The security library provides the primitives needed to compute the necessary cryptographic functions. The access to the HSM is realized by drivers which need to be specified in the services layer and in the Microcontroller Abstraction layer. Further the services layer (i.e. communication stack) needs to be modified in order to provide security services such as secure communication. A specification of the HSM interface will be added in the Microcontroller Abstraction layer. For that purpose, the EVITA software interface will use the “hook architecture” approach proposed by the *SeVeCom* software platform architecture **Fehler! Verweisquelle konnte nicht gefunden werden.**. This allows the integration of new security applications into existing implementations as AUTOSAR communication stack, which needs to be modified in order to reach security goals such as secure communication. EVITA further extends the foreseen AUTOSAR security library that provides well-defined access to the (basic) hardware protected security functionality. The security library enables and protects more complex security services required by e-safety applications such as entity authentication, communication protection, privacy management, or intrusion detection and response.

5 Outlook

The EVITA hard- and software security architecture will serve as basis for securing in-vehicular IT on-board networks. Formal verifications methods and tools will substantiate the development and design of the architecture and protocols. In order that the entire automotive industry can benefit from the EVITA project results, the specifications of the secure on-board architecture and communications protocols will be published as open specifications.

In order to evaluate the EVITA design, it will be implemented based on an FPGA and integrated within a V2X demonstrator. For prototyping, FPGAs will be used to extend standard

automotive ECUs with the functionality of cryptographic co-processors. The EVITA system will be tested and evaluated based on the EVITA architecture being integrated into a V2X demonstrator, showing e-safety applications based on V2X communication. Releasing the automotive HSMs for deployment in vehicles on public roads requires further implementation and testing efforts, which are out of scope of the EVITA project.

Acknowledgments

This work presents parts of the collaborative project EVITA co-funded by the European Commission under the 7th Framework Program.

References

[1] E-safety Vehicle Intrusion proTected Applications (EVITA) Project. www.evita-project.org, 2008.

[2] E. Kelling, M. Friedewald, T. Leimbach, M. Menzel, P. Saeger, H. Seudié, and B. Weyl: Specification and Evaluation of E-Security relevant Use Cases. Deliverable D2.1 of EVITA, 2009.

[3] SeVeCom Deliverable D1.1. VANETS Security Requirements Final Version. http://www.sevecom.org/Deliverables/Sevecom_Deliverable_D1.1_v2.0.pdf, November 2006.

[4] A. Ruddle, D. Ward, B. Weyl, S. Idrees, Y. Roudier, M. Friedewald, T. Leimbach, A. Fuchs, S. Gürgens, O. Henniger, R. Rieke, M. Ritscher, H. Broberg, L. Apvrille, R. Pacalet, and G. Predroza: Security requirements for automotive on-board networks based on dark-side scenarios. Deliverable D2.3 of EVITA, 2009.

[5] A. Fuchs and R. Rieke: Identification of authenticity requirements in systems of systems by functional security analysis. In *Workshop on Architecting Dependable Systems (WADS) at the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, Estoril, Lisbon, Portugal, 2009.

[6] O. Henniger, L. Apvrille, A. Fuchs, Y. Roudier, A. Ruddle, and B. Weyl: Security requirements for automotive on-board networks. In *9th International Conference on ITS Telecommunication*. Lille, France, 2009.

[7] Electronic Architecture and System Engineering for Integrated Safety Systems (EASIS) Project. www.easis-online.org, 2007.

[8] B. Schneier: Secrets and Lies. Digital Security in a Networked World. Chapter 21, Wiley, 2000.

[9] ISO/IEC 18045:2008: Information technology – Security techniques – Methodology for IT security evaluation. 2008.

[10] ISO/CD 26262: Road vehicles – Functional safety. ISO draft, 2006.

[11] ECRYPT II: Yearly Report on Algorithms and Keysizes (2008-2009), D.SPA.7 Rev. 1.0, ICT-2007-216676, 07/2009.

[12] NIST: Recommendation for Key Management, Special Publication 800-57 Part 1, 03/2007.

[13] T. Güneysu, C. Paar: Ultra High Performance ECC over NIST Primes on Commercial FPGAs. In *Cryptographic Hardware for Embedded Systems (CHES 2008)*, Washington D.C., USA, August 10–13), 2008.

[14] N. Pramstaller, C. Rechberger, V. Rijmen: A compact FPGA implementation of the hash function whirlpool. In *Proceedings of the 2006 ACM/SIGDA 14th International Symposium on Field Programmable Gate Arrays (FPGA 2006)*, Monterey, California, February 22–24,), 2006.

[15] M. Feldhofer, K. Lemke, E. Oswald, F.-X. Standaert, T. Wollinger, J. Wolkerstorfer: State of the Art in Hardware Architectures. Deliverable No. D.VAM2 – State of the Art in Hardware Architectures, September 2005.

[16] German Federal Office for Information Security (BSI): Application Notes and Interpretation of the Scheme (AIS) – Functionality classes and evaluation methodology for deterministic random number generators. December 2, 1999.

[17] Xilinx Inc.: MicroBlaze Soft Processor Core. www.xilinx.com/tools/microblaze.htm, 2009.

[18] G. Rouvroy, F. Standaert, J. Quisquater, J. Legat: Compact and Efficient Encryption/Decryption Module for FPGA Implementation of the AES Rijndael Very Well Suited for Small Embedded Applications. In *International Conference on Information Technology: Coding and Computing (ITCC 2004)* Volume 2, Las Vegas, Nevada, USA April 5–7, 2004.

[19] Automotive Open System Architecture (AUTOSAR). www.autosar.org, 2009.

[20] Secure Vehicle Communication (SeVeCom) Project. www.sevecom.org, 2006.