Risk-Oriented Security Engineering

Christof Ebert and Dominik Lieckfeldt [1]

Abstract: Virtually every connected system will be attacked sooner or later. A 100% secure solution is not feasible. Therefore, advanced risk assessment and mitigation is the order of the day. Risk-oriented security engineering for automotive systems helps in both designing for robust systems as well as effective mitigation upon attacks or exploits of vulnerabilities. Security must be integrated early in the design phase of a vehicle to understand the threats and risks to car functions. The security analysis provides requirements and test vectors and adequate measures can be derived for balanced costs and efforts. The results are useful in the partitioning phase when functionality is distributed to ECUs and networks. We will show with concrete examples how risk-oriented cyber security can be successfully achieved in automotive systems. Three levers for automotive security are addressed: (1) Product, i.e., designing for security for components and the system, (2) Process, i.e., implementing cyber security concepts in the development process and (3) Field, i.e., ensuring security concepts are applied during service activities and effective during regular operations.

Keywords: Cyber Security, Safety, embedded systems, quality requirements, risk management, validation

1 Introduction

1.1 Automotive Connectivity and Cyber-Security

More than 20 years ago, the invention of the CAN bus built the basis of connectivity. In the beginning, only two to three ECUs (electronic control units) were connected. But nowadays we have complex networks of sensors and actors with different bus systems like CAN, LIN, FlexRay, MOST or Ethernet. The interaction of functions in this distributed network is an essential part for our today's modern cars with all features for safety and comfort.

Besides the further development of innovative sensors like radar and camera systems and the analysis of the signals in highly complex ECU systems, the connected cars will be a driving factor for tomorrow's innovation. Internet connections will not only provide the need for information to the passenger. Functions like eCall or communication between cars or car to infrastructure (car2x) shows high potential to revolution the individual traffic. This includes the improvement of the traffic flow controlled by intelligent traffic lights, warnings from roadside stations or brake indication of adjacent cars. This builds the basis for enhanced driver assistant systems and automated driving. But the connection to the outer world bears also the risk for attacks to the car (Fig. 1).

[1] Vector Consulting Services, Ingersheimer Straße 24, D-70499 Stuttgart, E-Mail: Christof.Ebert@vector.com

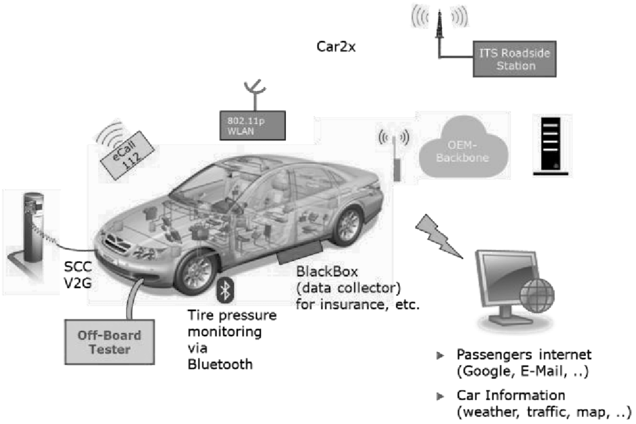


Fig. 1: Car with remote connections

The picture above shows several car connections that are already available today or will come up soon in the near future. Each connection to the car has a potential risk for an attack, regardless if it is wireless or wired. Just the threat is different. The access through a connector is only possible for a limited amount of cars, whereas a far field connection can be accessed from anywhere in the world. But also near field connections play an important role, such as tire pressure monitoring system, Bluetooth and wireless LAN. Security and reliability of these connections will be essential for the acceptance and success of these systems. With the introduction of this technology precautions must be taken to increase the reliability and to reduce the vulnerability to the system.

Obviously with growing connectivity functional safety needs security. Based on the specific challenges of automotive security, OEMs and suppliers have to realize an effective protection against manipulations of automotive E/E systems [EB2016]. Key points in the development of protected E/E systems are the proper identification of security requirements, the systematic realization of security functions, and a security validation to demonstrate that security requirements have been met. The following items need to be considered to achieve security in the car development process:

- Standardized process models for a systematic approach which is anchored in the complete development process. This starts on the requirements analysis through the design and development down to the test of components and the network.
- Quick software updates to close vulnerabilities in ECU software.
- Reliable protocols that are state-of-the-art and meet long-term security demands. Related to security this is often combined with cryptographic keys. So, a key management over the lifecycle of the vehicle must be maintained.

- In-vehicle networks and system architecture that provides flexibility and scalability and are designed under consideration of security aspects.

Based on our experiences in several client projects, we show which security engineering activities are required to create secure systems and how these activities can be performed efficiently in the automotive domain [EB2016]. In the following we want to take a view on each of these topics, what are the current activities, but also want to provide hints on how to mitigate the security risks.

1.2 Safety and Security

Night drive on the highway. The display suddenly flashes and the loudspeakers transmit a loud and painful sound. The driver is highly disturbed and tries to stop this annoyance. In doing so he is losing control over the car and causes an accident. Mere fiction? Not really. Continuously growing complexity within the electrical subsystems of the car, their interconnection by a variety of bus systems, and the use of standard components with open interfaces make networked systems within the car increasingly vulnerable. Such risks demand strong protection on various levels along the entire life-cycle of components and of the entire vehicle. Looking to past experiences with insufficient security in other domains, it is obvious that automotive security will determine which suppliers and electronic platforms (e.g. AUTOSAR) will capture the market for standard components. And it will determine how fast further communication systems (e.g., telematics with internet access) will be accepted by customers and policy makers.

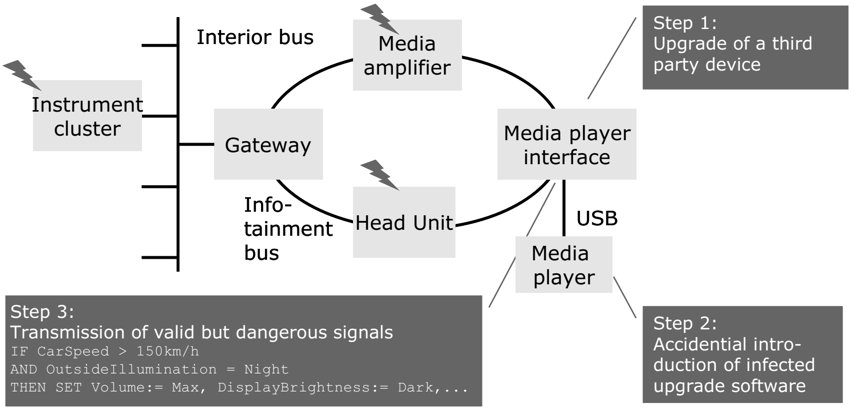
What is cyber-security? Basically, security is a quality attribute which heavily interacts with other such attributes, such as availability, safety or robustness. Security is the sum of all attributes of an information system or product which contributes towards ensuring that processing, storing and communicating of information sufficiently protects integrity, availability and trust. Security implies that the product will not do anything with the processed or managed information which is not explicitly intended by its specification. If for instance the classic definition of a functional requirement meant that the car can be started by turning the key, but would also allow a variety of mechanisms to start it otherwise, maybe for diagnostic or repair services (who was not in such situation that he needed support on the road and the person would open the trunk and start the engine directly?). Security implies that the car cannot be started except for the defined scenarios and is therefore protected against theft or misbehaviors. It's growing relevance comes from the simple observation that by defining functionalities alone, there is nothing said about the correlation of features, specifically if one of the many components of the car malfunctions. Many drivers of cars of the first generation of highly interconnected electrical control units distributed across the car will recall strange behaviors, such as windows which would open when switching on the radio. Automotive security has to ensure that any such malfunction or misuse case will not happen.

There is a big difference when we contrast safety and security. Safety is built upon reliability theory and looks into statistical malfunctions of components with small probabilities and how they will impact functionality. Security on the other hand has to deal with the worst cases with a probability of one because once known, they will be exploited. One might argue that safety is about criticality for the life and health of the system's user, while security is only about annoyances. It is however obvious that within a safety-critical system, such as a car, security meets safety because malfunctions can interact and cause disturbances that can result in accidents, as described in our introduction.

Vulnerability scenarios within cars have been changing fast over the past years. The increasing interconnection on different architectural layers (e.g., electrical control units, software components, configurations and their changes, communication inside and outside the car, diagnosis, telematics) has caused a level of complexity that was unknown so far. It is a mere question of time until the resulting loopholes and weaknesses are identified and abused. It was showed already that widely used automotive bus systems such as CAN and FlexRay can be brought from the outside – by connecting a device to any point of such bus systems – into overload conditions which will eventually cause malfunctions [EB2016].

State of the art communication systems increasingly offer open interfaces (e.g., DVDs, E-Mails, USB, Bluetooth, IP-based diagnostics) that allow to inject viruses and Trojan horses to the respective embedded operating systems. Also, defective code and configuration settings can create new and unknown vulnerabilities as we are used to from many information systems.

This is what drives the security attacks in our illustrative case study from the beginning of this article. Fig. 2 illustrates the primary sequence of incidents that caused the flashing display and the loudspeakers or head unit to transmit such loud signals.



Step 1:
Upgrade of a third party device

Step 2:
Accidential intro-duction of infected upgrade software

Step 3:
Transmission of valid but dangerous signals

```
IF CarSpeed > 150km/h
AND OutsideIllumination = Night
THEN SET Volume:= Max, DisplayBrightness:= Dark,...
```

Fig. 2: „Night Drive" – How could it happen?

As so often in security attacks, the first step was just a normal upgrade of the multimedia equipment with a better device. Needless to say that it was not delivered and installed by the OEM, but came through an internet delivery for perceived cost reasons and enhanced functionality. In a second step infected software came into the multimedia devices, probably via an infected USB stick or from a media file. It could well be that software upgrades to one of the media devices also brought this infected software into the system. From here onwards it was just normal cause and effect, namely transmission of valid but dangerous signals on the infotainment bus which were triggered by listening to signals with speed and outside illumination. These signals are almost omnipresent in car networks due to many dependencies on these factors.

The different reasons for insufficient automotive security are illustrated with examples in Fig. 3. They are distinguished according to the different scenarios that cause the vulnerability or security problem within the product (e.g., functionality, architecture, configurations), in the process (e.g., design, development, validation, stakeholder communication), and in the field (e.g., maintenance, enhancements, diagnosis).

All these scenarios result from unawareness of security needs and security technology, be it by ignorance (“this won’t matter in cars because all our critical electrical units are protected by cryptography”), arrogance (“security matters only in information systems”) or naivety (“we have verified all requirements and components according to our established test strategy”). An overall security strategy is mostly missing.

Causes	Product / architecture	Development process	After-sales / field
By ignorance	- security is insufficiently supported by architecture - components are individually verified but not the system in which they operate	- missing security requirements - no abuse and misuse scenarios - insufficient verification during the entire component and system life-cycle - inadequate checklists and design guidelines to design for security	- changes and modifications are not sufficiently validated against security requirements and abuse / misuse scenarios - security impacts of changing a component or introducing a new version or variant of a software / hardware component are not analyzed
By arrogance	- experiences from other products, domains and markets (e.g., IT, telecommunications) are not considered - security is designed only on the basis of firewalls, gateways and protected components	- inadequate or missing communication between automotive electrical engineers and IT security experts - development processes specifically on system level without security requirements and checks	- new software releases are introduced during production and after-sales without considering security checks and qualifications
By naivety	- insufficient training on security for software and systems engineering - unknown misuse / abuse scenarios - automotive bus systems are active even when ignition is off	- unknown state of the practice for security verification tools and test methods - missing requirements and criteria for security - non-proprietary or open components are introduced without verifying security criteria on system level	- software components or media in use which have not been qualified - after-sales devices and components are integrated to the car without assessing the impacts on other subsystems such as network overload

Fig. 3: Causes for security vulnerabilities and issues in the life-cycle of a car

Typically, components are individually protected such as encrypted flashware for an engine controller. Critical functionality such as engine management, theft protection or engine diagnosis is hardened and verified. Increasingly secure networks and architectures are discussed and will certainly influence the design of cars ten years from now. Safety has received a lot of focus recently in automotive engineering and qualification of components and systems, such as processes to ensure proper handling and engineering according to SIL-levels. But safety and associated design rules are insufficient as we have learned before. They look to faults and their probabilities, while security has to deal with the worst case in scenarios where a probability is replaced by the willingness of the attacker to cause the worst possible damage.

Two aspects related to security in embedded systems have to be considered: (1) Attack scenarios go well beyond individual components and functions. (2) While safety deals with avoiding critical failure modes, security has to cope with intelligently introduced causes of faults, which is far more difficult, given that the attackers' intelligence, willingness, determinedness, and creativity often exceed that of the engineers looking to a problem from the – different – perspective of how to solve it, and not how to find loopholes and strange feature correlations.

Telecommunication and information systems have realized several years ago that isolated mechanisms (e.g., distributed functionality in proprietary subsystems, protection on component-level, gateways and firewalls between components, validation of critical functions) are insufficient. This article underlines together with concrete examples how automotive security can be achieved. We will take the three different perspectives that were introduced in Fig. 3, namely (1) the product and its architecture (e.g., specification of security requirements, misuse and abuse cases, vulnerability analysis, inherently secure architectures); (2) engineering for security during the development process (e.g., FMEA and hazard analysis as a basis for security, protection on component- and on system-level, systematic verification, code analysis, validation on product-level); and (3) the relevant after-sales activities in the field (e.g., fault analysis, patch and correction handling, emergency response and handling, distribution of corrections and protective mechanisms).

Security and related measures demand well-founded concepts all along the life-cycle of both components and the car itself, especially if their effectiveness has to be proven at a later point due to legal actions. With this article we strive not only to provide guidance for specific misuse cases but to change the mentality of engineers of embedded systems towards designing for security – rather than for functionality.

1.3 Risk-oriented Security

Developing secure software is challenging for several reasons, namely because increasingly systems are connected, most software is developed in a global context in heterogeneous teams with various skills, systems complexity is exploding with embedded and IT systems converging such as IoT, and both budget and cycle times are continuously decreasing. For instance a modern car has almost hundred embedded microcontrollers on board and is connected over several external interfaces to a variety of cloud technologies. At the same time cyber-attacks and vulnerabilities are increasing. Therefore, software technology and the underlying security engineering have to be constantly improved.

While there is a movement towards better understanding security from the ground up, many of the existing approaches in managing security have been focused around encryption, developing malware software, and to detect attacks to networks and systems. Existing methods and tools are limited by large number of false positives and inability to consistently trace such issues to the root causes. In this article we will particular draw attention to all aspects of security from specification to design and life-cycle support.

Over the past decade trends like connected car and driver assistance systems among others have led to software and connectivity playing an increasingly important part in developing vehicles and also for business models of OEMs and suppliers likewise.

Devastating impact of security issues is already known from industrial sectors like IT-infrastructure, aviation, information technology and telecommunications, industrial control systems and energy and financial payments. Virtually every connected system will be attacked sooner or later. A 100% secure solution is not feasible. Therefore, advanced risk assessment and mitigation is necessary to protect assets. Consequently, the typical solution to security in these industries relies on suitable risk assessment that projects threats on assets of interests. Thereby cost of implementing specific security measures can be compared with the probability of a particular threat that they counter.

Asset-based risk assessment is a suitable tool for companies to steer efforts for security engineering in a systematic and comprehensive way and thereby involve all relevant stakeholders in the organization. For example, a CEO may not find it very helpful to have a long exhaustive list with every attack vector or potential threat – they need to be provided with a ranked listing and useful decision-support tools which clearly shows alternatives and consequences. From the view of an automotive system developer, a flat listing of potential threats might not help to improve the system. To really help, they need to be able to map security threats, countermeasures and requirements to system/architecture elements in their scope of the project.

The systematic management of security threats and associated security goals is essential to actually providing safe and competitive products, and to protect valuable assets and business models.

But what makes security engineering so complex? Automotive developers face the challenge of securing a system against attackers whose capabilities and intentions are at best partially known. Some attacks might today appear infeasible, but todays impossible attacks might become more likely in the near future. An example of this is attacking a vehicle simply by exploiting wireless interfaces, 20 years ago would have been extremely unlikely, however today a cheap software defined radio and accomplish these types of attacks with little effort. On the other hand, an attacker might invest more effort into launching an attack the more valuable a successful attack is to him. Some attacks represent more effort to the attacker than others given the specific potential of the attacker. It is this risk/reward payoff that is analyzed in security engineering. Likewise during testing and verification, suitable methods to verify that the vehicle has the required security level and process goals like, test strategy and coverage, need to be chosen.

Furthermore, the assets to be protected from attacks are decided by stakeholders involved, e.g. drivers would indicate different assets of their vehicle to be protected compared with what an automotive developer considers an asset. However, customers/drivers need to be satisfied with their vehicle in order to buy another one from the same company. Consequently, security engineering must seek tradeoffs between cost of security measures and benefit to assets in order to make sustainable decisions.

Security concepts must balance the cost of not having enough security and thus being successful attacked with all damaging consequences and the cost spent to implement appropriate security mechanisms and keep them updated along the life-cycle of the car – well beyond end of production. We therefore introduce here a strict risk-oriented approach to security (Fig. 4).

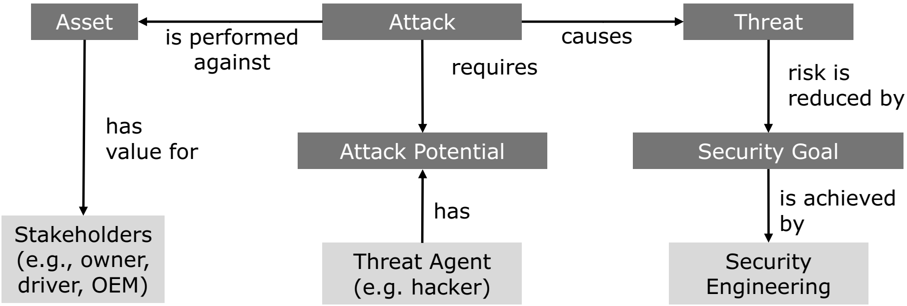


Fig. 4: Overview of risk-oriented cyber-security analysis process with the major steps of asset determination, attack potentials and security goals derived from threats

To summarize, the relationship between assets, attackers and threats is complex and dynamic (e.g. attacks are more probable the less effort is required and the more value successful attacks represent; attack vectors and effort change over time). Furthermore, common understanding of assets among all stakeholders of security engineering is mandatory in order to provide information for steering the security engineering.

Choosing the right set of security engineering methods for analysis, concept and testing is challenging but required in order to enable goal-oriented and manageable security engineering.

Risk-based Security Engineering combines state-of-the-art methods for automotive security risk assessment in a practical framework and supports all involved stakeholders to develop “secure-enough” products. The method and our approach for proposing a concrete technical security concept is based upon security best practices such as:

- SAE J3061 (Automotive cyber-security) being the first standard on the topic of automotive cyber-security but also being aware that it primarily enriches ISO 26262 towards security for functional safety [SAE2016].
- ISO 15408 (Evaluation criteria for IT security) with its focus on IT systems, specifically the 7 evaluation assurance levels (EAL) for security requirements and guidance on common criteria a standardized practice translated by Vector to automotive common criteria.
- ISO 27001 (Information security management systems) with its governance requirements for security engineering across the entire value chain.
- IEC 62443 (Industrial communication network security) with its strong view on distributed systems and necessary security technologies and governance.
- ISO 26262 (Automotive functional safety) using its clear focus on automotive electronic systems with good coverage of entire life-cycle; revision in 2016 [ISO2017].
- IEC 61508 (Functional safety for electronic systems) while being aware that it is only a high-level functional safety guidance for electronic systems.

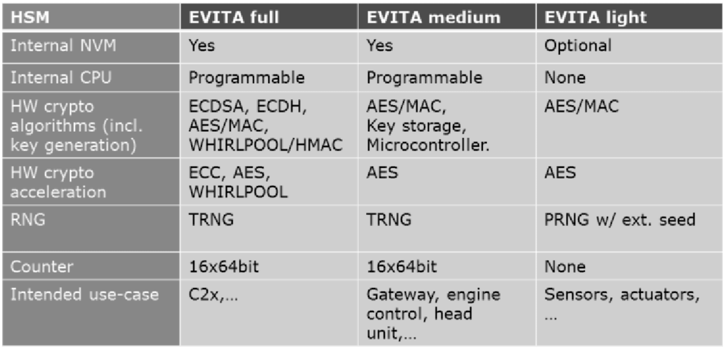
Our Vector Security Check and underlying security engineering methods have adopted the state of the practice in security evaluation and proposed mitigation [EB2016]. It is using significant research work from our worldwide security projects. It also uses external best practices, such as “E-safety vehicle intrusion protected applications“ (EVITA) funded by European Union [Ev2017], HEAVENS [Is2014], and other proposed methods for security risk assessment in automotive development [Se2017, Si2017, Pr2017, ETSI2010].

We will furtheron show by examples how to use the risk-oriented security concept covering the entire security life-cycle with focus on the upper left activities, namely

- Asset Definition and Threat and Risk analysis
- Security Goals
- Security Concept

1.4 Related Work in Automotive Security

The automotive industry is already engaged in security topics since several years. Several (EU-) funded projects had been launched for researches on Car2x. In the following a few of them will be presented. The SEVECOM project (www.sevecom.org, [Se2017]) has analyzed risks and threats and has defined first general security architecture. A notably project for security was EVITA ("E-safety vehicle intrusion protected applications", www.evita-project.org, [Ev2017]; Fig. 5). The main objectives were to design a secure on-board network and the definition of building blocks to protect security relevant components and data inside a vehicle. One of the major outcomes was the definition of a hardware security module, defined in three versions: light, medium and full. Each version requires at least a hardware acceleration for data encryption (AES), secure key storage and a secure boot. These requirements show equivalences to the SHE (Secured Hardware Module) defined by the HIS (Hersteller Initiative Software, www.automotive-his.de).



HSM	EVITA full	EVITA medium	EVITA light
Internal NVM	Yes	Yes	Optional
Internal CPU	Programmable	Programmable	None
HW crypto algorithms (incl. key generation)	ECDSA, ECDH, AES/MAC, WHIRLPOOL/HMAC	AES/MAC, Key storage, Microcontroller.	AES/MAC
HW crypto acceleration	ECC, AES, WHIRLPOOL	AES	AES
RNG	TRNG	TRNG	PRNG w/ ext. seed
Counter	16x64bit	16x64bit	None
Intended use-case	C2x,...	Gateway, engine control, head unit,...	Sensors, actuators, ...

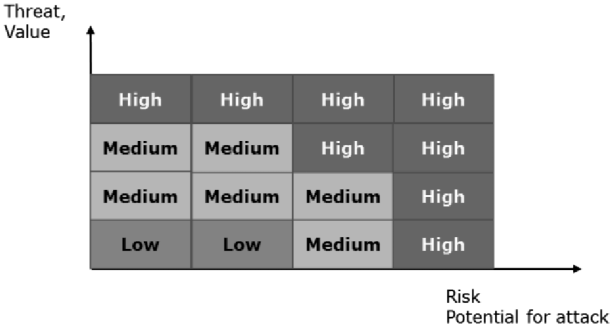
Fig. 5: EVITA classification for the hardware security module (HSM)

In-field tests for a Car2x communication was made in the Sim project (www.simtd.de, [Si2017]). In the area of Frankfurt, a field test was established with more than 100 test vehicles. Highways, country roads and city traffics were equipped with infrastructure to communicate with. A currently active project is PRESERVE (http://www.preserve-project.eu, [Pr2017]). The main objective is the design of security architecture for vehicle-to-infrastructure (V2x), to setup and test the system. This shall be achieved by setting up a fully operating security subsystem in a real environment with consideration of cost and performance. This includes also a further hardware environment with adequate performance.

These activities are important preconditions for a secure communication that is standardized in the Car2x area. This is essential for interoperability between cars from different car manufacturers and beyond national boundaries. In Europe there is the CAR2CAR consortium working on standards for vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and a cooperative intelligent transport system (C-ITS).

2 Security analysis

Security in a complex system cannot be achieved by applying countermeasures on single items. It requires an analysis of the complete functionality or system as a whole and to apply countermeasures as an integral part. First, you need to identify what are the assets I want to protect. Besides financial aspects also confidentiality and, especially for the automotive industry, safety functions must be considered carefully. The next step would be a threat analysis: who has access to my assets, what are potential attackers and where are my access points. A typical approach to this is the construction of a data flow diagram in which the assets are identified. It provides an overview of all connections and access points, where attacks and manipulations can be achieved. From the material above a risk assessment can be done to obtain the measurements and results in a classification of the risk. An example of such a risk assessment can be found in the picture below. Here, as an example, the classification was defined in three categories: Low, medium and high (Fig. 6).



High	High	High	High
Medium	Medium	High	High
Medium	Medium	Medium	High
Low	Low	Medium	High

Fig. 6: Definition of security level derived from threat analysis and risk assessment.

This process provides systematic means to deal with the subject and results in a balanced trade-off for cost and efforts. Depending on the determined security level countermeasures can be defined on system level and further derived as security input requirements for ECUs. The analysis phase provides now also requirements for hardware extensions of the ECU, e.g. if hardware acceleration is needed for authentication or if a specific key management is required for higher security measures. The requirements are also an input to define test vectors on functional (for an ECU) and system level (for the vehicle). These tests, together with standard penetration tests, then will help to provide evidence for successful application of the security to the function and system.

3 In-vehicle security

The IT industry deals already since years with strategies for data protection and to provide secured networks to prevent them against unauthorized access. Wide experiences are available here, that, with special considerations, can be adapted and are useful for the automotive industry as well. Similar activities can be seen such as the adaptation of the Ethernet when BroadR-Reach was introduced to the automotive area. This allows also taking over the proven software architecture of Ethernet, so that a number of approved protocols are available as well for a secured data transmission. Essentially, they are based on cryptography, software algorithms based on more or less complex mathematics. The algorithms itself are not the secret and are available to the public but keys provide the secret and they must be created, distributed and maintained carefully. A popular key management system used by the IT industry is the PKI (Public Key Infrastructure). It contains a hierarchical certificate management with associated keys and builds the basis for an authenticated communication between partners.

3.1 Security Engineering

While security requirements are concerned with what has to be protected, security engineering defines how the protection is realized. It affects all activities that are associated with "normal" engineering, such as system design, software construction, tests and after sales. We regard each of these activities in the following paragraphs.

In a recent client project, we had to identify and prioritize security requirements for an auto-motive E/E component. Based on the size and complexity of the component development project and the given capacity for security engineering, we selected an agile approach [EB2016] that was conducted in form of several workshops. The client's engineers provided expertise on the component's functions and their implementation. We moderated the workshop and provided expertise in security requirements analysis as well as knowledge on the used technologies' vulnerabilities and sensible protection mechanisms.

For better understanding we will show some hands-on examples from current security projects:

- Adapt the development processes to factor in security engineering activities. Security engineering activities are known, scheduled, and executed smoothly within the "normal" development, not in an ad hoc way. Security is considered from the beginning on through the complete project. Additionally, synergies can be exploited (e.g. a configuration management process can prevent quick fixes that have not been tested against security vulnerabilities).
- Systematically elicit security requirements. Elements that have to be protected are known from the beginning on, allowing for stringent realization of their security. Additionally, security requirements can be used to deduce test cases for security validation.

- Thoroughly review or test any security relevant arte-fact. Reviews of security engineering artefacts such as security requirements and security concept as well as simulations of security functionalities and code analyses allow for the identification of vulnerabilities at the earliest possible time.
- Use analysis and test tools. Automated tools reduce effort and allow for efficient and comprehensive analysis and (regression) testing. For instance the Vector PREEvision PLM and modelling environment provides a strong collaborative engineering backbone for ensuring application of above measures along the lifecycle.
- Manage embedded security competencies. Many activities of security engineering require a specific embedded security expertise, e.g. identification of vulnerabilities, design of the security architecture, secure implementation, performance of security tests, and review of security-related work products. Without this expertise, effective security engineering is near to impossible. Therefore, build up embedded security competence in your organizational unit or obtain it from internal or external providers.

We do not claim that these are the only valid solutions. Depending on e.g. corporate culture, existing experiences, and project size and complexity, other solutions may be preferred. Independent of the approach used, we noticed several activities that benefitted the introduction and the performance of security engineering in general [EB2016].

3.2 Software update and maintenance

To enable efficient after sales activities in spite of constraining security mechanisms, several aspects need to be addressed. How can software updates be performed in the field with both security against unauthorized manipulations and justifiable logistical effort? The association of German car manufacturers (HIS) has created specifications for secure flashing of ECU software, for which conforming flash bootloaders are available. However, the concrete realization of the related logistical infrastructure needs to be considered (Fig. 7).

An important aspect of after sales activities is the way OEMs and suppliers react when a security issue is detected in a fielded vehicle. Such scenarios have to be foreseen before the vehicle’s SOP and procedures that define actions and responsibilities have to be set up. Actions to be planned are risk assessment of the issue, elimination of critical software vulnerabilities, and update of the software in the field. To achieve an efficient issue handling, a smooth cooperation between OEM and suppliers is required.

Certificates are building the basic concept for a secure and authenticated communication between the vehicle and the backend. They are managed in a PKI system installed and maintained by the OEM. This allows customer oriented service and maintenance with online connections to the vehicle. In case of a car problem, first diagnostic analysis could be made by the OEM help center in case of a malfunction. It also can be used to report early recognized anomalies and with the collection and analysis of further data an early warning could be sent to the driver before a harmful damage occurs.

Optimal preparation of a service can be achieved when the car workshop reads out online car information. Finally, software updates can be initiated quick and easy without the need to enter the workshop. This can be particularly helpful if software problems are encountered. This is an important if not even an essential pre-requisite for connected cars and provides several advantages: if an attack has been, it can be quickly closed by a software update. Like on a PC, software patches and updates can be distributed to close the vulnerabilities. A secure vehicle-to-backend connection can also provide a secure way to communicate with the internet. It opens also the possibility to perform software updates to ECUs if, for example, a critical software failure in an ECU was encounter.

The advantage of software maintenance over the backend of the OEM is obvious. The communication between the two systems can be strongly restricted, so that other connections are simply not possible, because the firewall at this location does not need to allow any other accesses. The connections are restricted to those partners who have access to the keys and certificates. In addition, system information can be gathered in the field and (anonymously) transmitted, which helps improving car functions. Moreover, this can open new business divisions for an EOM like cloud-services, function enabling, secure internet access or software-as-a-product.

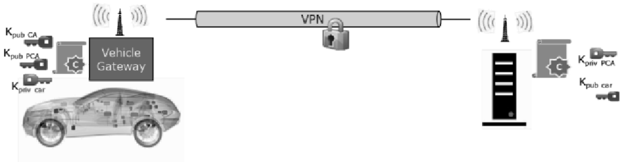


Fig. 7: Secured Remote-connection of the vehicle to the OEM backbone using certificates.

4 Case Study: Connectivity

A major objective in the IT industry is the provisioning of high performance and secure networks in enterprises. The location of the items on the network is just one aspect for the organization and operation of such a network. Considering security aspects in the basic structure from the very beginning can provide essential advantages for the flexibility and scalability of such a network. It can also reduce the risk for attacks. The major attack scenario sin automotive vehicles are described in Fig. 8

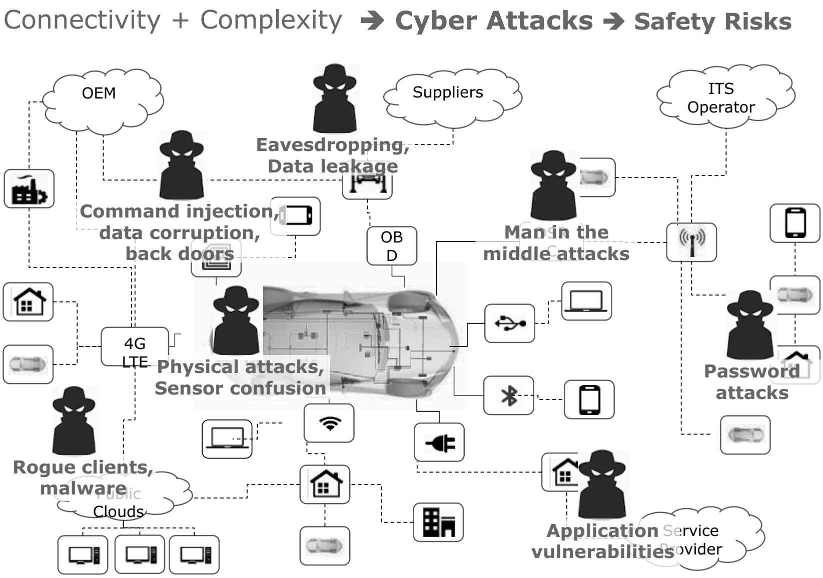

Connectivity + Complexity ➔ Cyber Attacks ➔ Safety Risks
OEM
Suppliers
ITS Operator
Eavesdropping, Data leakage
Command injection, data corruption, back doors
OBD
Man in the middle attacks
4G LTE
Physical attacks, Sensor confusion
Password attacks
Rogue clients, malware
Public Clouds
Application vulnerabilities
Service Provider

Fig. 8: Increasing connectivity drives complexity and enables multiple attack paths.

Security components like firewall and router are also important parts and are useful to separate networks. For example, account computers will not be connected to the same network as that from marketing or development. Instead, computers are grouped to separate networks depending on their use case and traffic. A router interconnects the networks and provides data exchange between them. It passes only the relevant and allowed data from one network to the other. The access to computers is already restricted by the structure of the network. The router with an integrated firewall also manages the access to the internet. This device observes the incoming and outgoing traffic and can be configured that only allowed traffic will pass. Additionally, maintenance can be done easily on the central part by applying patches or re-configure the device if needed.

Let's consider a car network under the aspects shown above. At the very beginning, a safety and security analysis has been performed and the networks are partitioned so that the connected items are grouped under functional, safety and security aspects. The different networks are interconnected by a gateway. The ECU with a remote connection is considered particularly as unsafe. Even if great care was taken during the software implementation, a failure cannot be excluded and the potential risk is too high that someone could capture and take control of the ECU from outside. To minimize the risk, this ECU should not have access to any other internal networks. We locate this function into a separate ECU (inter comm. module) and connect it through the gateway (Fig. 9).

The gateway can now contain a firewall that has separate filter rules for each subnet. Only those messages are passed to other networks that are allowed. The traffic inside a network is not restricted and affected by the firewall.

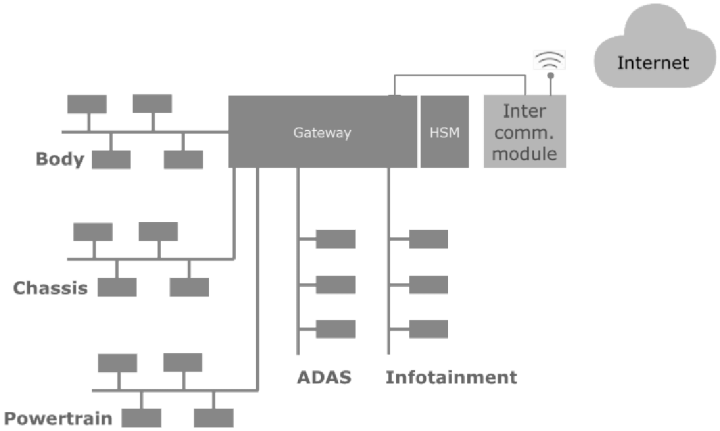


Fig. 9: Vehicle network with remote connectivity and gateway

We now take advantage of our threat analysis and risk assessment that has provided a detailed security analysis of our system. In the data flow diagram we saw the interaction of signals between partitioned functions separated in ECUs. This has already helped us to separate the ECUs to different networks, respectively partitioned the networks according to safety, security and functional aspects. We can now classify networks into security zones according to the safety and security requirements of the transmitted signals. If a network mainly contains signals with high security and safety requirements, that is classified as high security zone. The network with intermediate safety and security data is classified as a medium security zone. The network that contains just a few signals with safety and security requirements and many signals from remote connections is a low security zone. A network that contains an ECU with a remote connectivity must be treated as unsafe in principle and is therefore in a low security zone or even completely isolated (Fig. 10).

The origin and distribution of the signals influences the settings of the firewall in the gateway. The presence of signals from other security zones gives an indication to the security measures for the internal signals. If a network is physically isolated and signals from other networks are rarely used, the threat potentials are low. Unless other threats from adversaries[2] are identified, reduced measures can be applied for signals in such a security zone. This reduces efforts and costs for security measures of these ECUs. It shows how partitioning provides advantages. The signal flow from high to low security zones is not critical. However, if threats for data manipulation on the network are given, security measures like authentication or confidentiality can be added to the data. Greater care must be taken in the other direction. The risk potentials for these signals must be observed carefully. Also, if filter rules of the firewall can influence the complete security zone settings. For counter measures, authentication on signal may be required.

[2] For example, the manipulation of sensor signals for the engine control on the powertrain.

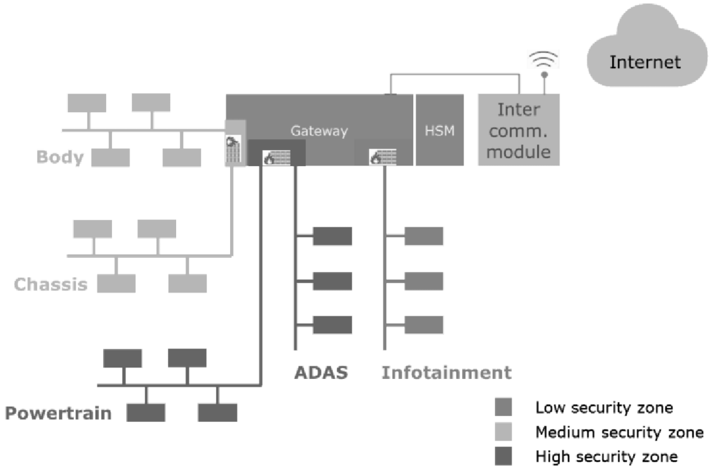


Fig. 10: Security zones in automotive networks

5 Conclusion

Automotive security has gained huge relevance in short time-frame. Attacks are reported today almost continuously and therefore systems must be protected and hardened. Safety needs security as a mandatory condition, which means that any safety-critical system as a minimum must also be protected for cyber security. Security must be integrated early in the design phase of a vehicle to understand the threats and risks to car functions.

Risk-oriented security helps to balance growing security threats with increasing complexity over the entire life-cycle. Unlike many previous attempts our research and many practice projects indicate that while design for security is good, it is not good enough. Effective security must handle the entire life-cycle (Fig. 11).

Security analysis provides requirements and test vectors and adequate measures can be derived for balanced costs and efforts. The results are useful in the partitioning phase when functionality is distributed to ECUs and networks. Networks isolated under security aspects helps to reduce the risks and efforts. Security key management will become an important part and requires a key infrastructure (PKI) managed by the OEM over the production and maintenance phase of the vehicle. Additionally the secure key handling inside an ECU and the usage in development, production and maintenance phase must be considered. The PKI must be online to allow access by the workshops. Additionally, an OEM backend is needed that allows flash programming over the air, at least to provide hot fixes and patches. Such a backend can provide additional security and features to the car owners, but can also open new business divisions for OEMs.

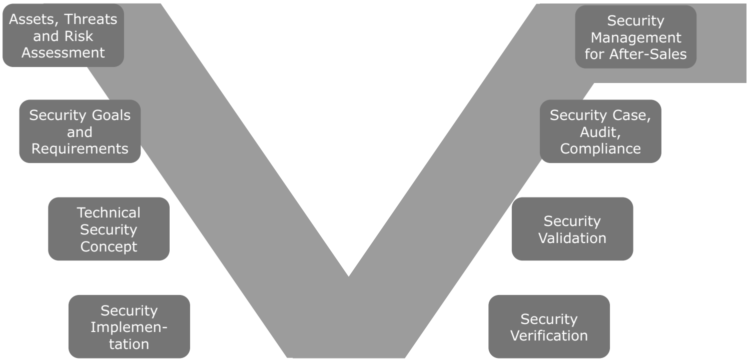

Assets, Threats and Risk Assessment
Security Goals and Requirements
Technical Security Concept
Security Implemen-tation
Security Verification
Security Validation
Security Case, Audit, Compliance
Security Management for After-Sales

Fig. 11: Security Engineering along the Life-cycle

Companies urgently need to build up necessary basic security expertise and obtain adequate external support, specifically where security meets safety. Mature development processes provide a good basis but need to be amended with dedicated security engineering activities as we have showed in this article.

Bibliography

[EB2016] C. Ebert, A. Braatz: Automotive security engineering, Vector White Paper 2016.

[Se2017] SeVeCom (Secured Vehicular Communication) project: www.sevecom.org, Last accessed on 12.Mrc.2017.

[Si2017] SIMTD (Secure Intelligent Mobility): www.simtd.de, Last accessed on 12.Mrc.2017

[Pr2017] PRESERVE (Preparing Secure Vehicle-to-X Communication Systems): www.preserve-project.eu, Last accessed on 12.Mrc.2017

[Ev2017] EVITA (E-safety vehicle intrusion protected applications): www.evita-project.org, Last accessed on 12.Mrc.2017.

[ISO2017] ISO 26262, ed.2, draft - Road vehicles — Functional safety, Last accessed on 12.Mrc.2017, www.iso.org

[ETSI2010] ETSI TR 102 893, "Intelligent Transport Systems (ITS); Security; Threat, Vulnerability and Risk Analysis (TVRA)," 2010.

[Is2014] M. Islam et al.: Project overview HEAVENS - Healing Vulnerabilities to Enhance Software Security and Safety, Volvo AB, 2014.

[SAE2016] SAE International: “Cybersecurity Guidebook for Cyber-Physical Vehicle Systems, J3061_201601”, 2016, www.sae.org