# Embedded Security recap

Thilo Schumann, CAN in Automation

**Cybersecurity is getting more attention as devices, systems, companies are getting hacked. At previous iCCs different solutions have been presented, which solve singular problems. Different standards have been developed or are still work in progress like ISO 27001 (IT), IEC 62443 (industrial) and ISO 21434 (automotive). Those standards do not specify any technology as such to solve the problem of cybersecurity but define processes and procedure to classify security threats and how to cope with it.**

## Introduction

Previous presentations and abstracts on the topic of cybersecurity in embedded systems are either fall into the category of focusing on one technical solutions solving a singular issue, or breaking the security of a particular embedded solution. Especially the later show that cybersecurity is not a singular technical issue, but needs to be seen in a broader sense.

The need for cybersecurity comes from the thinking that there are no trustworthy parties. In contrast embedded systems are designed that those are isolated and every participating party is trustworthy.

## Standardisation

There needs to be a shift in the general thinking of system integrators and device designers for embedded systems. Solutions on many different levels are required. There is international standardisation going on to address that topic. In the IT industry the relevant standard is ISO 27001 series, in the industrial world it is IEC 62443 series, and in automotive it is ISO 21434 series.

At various degrees common to these standards is to view the system in general, to consider the whole lifecycle of the system, and to consider and define different threat models with different bad actors and stakeholders in mind.

## Life cycle

Life cycle management plays a very important part of todays cybersecurity. An embedded system is developed with the use and application of existing standards. Those standards are defined by unknown parties. Development is based on sourcing existing solutions either as software and hardware. Those solutions are developed by unknown parties. Production is outsourced to third parties. Those production facilities are partly unknown, or source their materials from unknown parties. Embedded systems are then used as part of their use case by unknown parties. Embedded systems are updated during usage to either fix bugs or enhance functionality. Those update processes are done via unknown parties. After use the system is decomposed and maybe recycled by unknown parties. At almost any of those different stages different stakeholders and bad actors have different interests.

## Threat model

Threat modelling is a technique to grasp the different stakeholders and bad actors with their different interests.

For a singular threat model the important assets for a singular stakeholder needs to be identified. Then the potential bad actors for that asset are identified and classified according to their potential skills. The problem in cybersecurity is, that there is not only one single threat model for a particular embedded systems but many different.

## Threat model tachograph

For example the electronic tachograph that is required according to EU law originally issued in 2002 and required by all electronic tachographs since 2006 required a secure communication from the sensor to the tachograph. This secure communication is standardised in ISO 16844.

The important asset had been identified, which mainly meant speed and distance measurement. The communication between the sensor and the head-unit need to be secured. The data in the head-unit need to be secured as well. The stakeholder here is the government and law enforcement. The bad actors are drivers and logistic companies, because they may have a monetary interest in driving faster than allowed, which gives them a competition benefit. The technical expertise of those bad actors where considered modest, in terms they may buy in expertise, technical modifications to circumvent the security. To date known, the communication seems to be secure. But the storage of the data in the head-unit is the weak point as recent development had shown. Bad actors can easily get hold of devices that are able to manipulate the data. Cybersecurity is not a singular topic.

## Stakeholders

Every stakeholder has their own interest. Some of those interests may contradict each other. Especially when cybersecurity is combined with privacy and monetary interests, commonly known as: „Who owns the data?“

A device manufacturer may has interest in the logging data of its devices. But when he sold the device to the system manufacturer does it still belongs to the device manufacturer. This also applies when the system manufacturer sells the system to the user, when the user sells the system for second use, or decommissioning and recycling. There are lawyers involved in those discussion. But all of this comes back to possible and impossible technical solutions on the implementation side, especially by designing and defining the threat models.

## Secure communication in CAN, CAN FD, and CAN XL

CAN itself, like many data link layer protocols, was not designed with secure communication in mind. Secure communication is a new requirement as of late. Some secure communication standards for CAN had been developed like ISO 16844, and Autocar SecOS and are in use today. These standards are providing some kind of security. This security has to be established in a secure environment. In the case of ISO 16844 these are certified shops. During the establishment of the secure communication the participating devices are linked together. When this link goes down for whatever reason it has to be re-established in the certified shop otherwise it is considered as compromised. That means there is no plug and play for the end user.

## Attack vectors

The problems is CAN is a bus oriented network with so-called multi-master capability. That means, every device on the network is allowed to communicate at any time. There is also no verification mechanism. Every device is trusted in the same way. A malicious device cannot be distinguished from any other legit device. A malicious device may also be a legit device with a re-programmed firmware to make it malicious.

For that reason many different attack vectors need to be distinguished. The lowest level is an attack on the CAN wires. By controlling the IO pins of a micro-controller directly attached to a CAN transceiver arbitrary messages can be crafted[1]. That means that original messages may be corrupted and overwritten. Overwriting is realised in terms of that the malicious message may follows immediately on the original message in terms of timing. On the receiving devices the malicious data overwrites the original data.

The same can be applied even without direct access to the pins but to the CAN controller. But here it is much more complicated to apply correct timing to realise the same behaviour.

The problem in general is: direct access to the wires is required either by applying a dedicated malicious device or by re-programming a legit device with a malicious software. But the later can be done remotely by any means of software update. With Internet connectivity the Internet gateway becomes the critical part for remote attacks. The Internet gateway needs to implement a firewall. The Internet gateway needs to monitor the traffic and needs to verify software updates.

### Counter measures

Currently there are many different strategies in development to provide countermeasures. A countermeasure against malicious messages from an unknown device or re-programmed legit device is, that the CAN transceiver in each and every device may maintain a black and white list of CAN messages[2]. The CAN transceiver lets through only legit messages according to its internal white list. If the software of the device is maliciously re-programmed, the malicious messages are blocked directly by the CAN transceiver. This has some benefits and some tradeoffs. The tradeoff is that the system designer needs to know fix the network design in the sense the message flow between device is fixed. Dynamically assigned CAN messages are not possible in the scenario either for plug and play systems or for later feature upgrades. If this needs to be provided the white and black lists in the CAN transceiver need to be upgradeable. Upgradability means, a malicious software upgrade can also upgrade this white and black list.

Another counter measure is to secure the communication between CAN controllers by applying authenticity or encryption[3]. A secret key between two devices is chosen or pre-setup and then those two CAN controllers can securely exchange messages. The tradeoff is, that this can be done between two arbitrary CAN controller who know about the protocol and are able to follow the protocol. An additional certificate infrastructure is required to ensure real authenticity. Another tradeoff is that the communication between two device directly attached to the same CAN is secured. It is no end-to-end security through gateways. The malicious device may be the gateway.

Another counter measure is to provide end-to-end security by applying and implement principles defined in well established standards like Transport Layer Security (TLS)[4]. End-to-end security is very well suited for transmitting singular data end-to-end. The tradeoff is that it does not allow broadcast. In that sense it is very well suited for software download/update, but not for broadcasting control data. Also an additional certificate infrastructure is require to ensure real authenticity.

### Certificate infrastructure

Many implementations of secure communication require authenticity. Authenticity means each device can prove to each other device that it is legit. That means certificates need to be exchanged that require a provable chain up to the root. The general problem of a certificate infrastructure is, that each an every device not only need a unique serial number but the manufacturer need to maintain individual certificates for each and every device produced. The attack surface extends that certificates of defunct devices may be re-used for malicious devices. That requires that certificates need short term validity. Short term validity requires mechanisms to update certificates before certificates expire. Certificate infrastructure implementation and maintenance is expensive and difficult[5].

### Summary

Cybersecurity in embedded systems and embedded networks cannot be treated as a singular problem with a singular solution. Cybersecurity needs to be considered from the overall view. ISO 21434 in the automotive world try to provide an overview without limiting dedicated technical solutions. The standard tries to grasp the complete picture of cybersecurity by not focusing on a singular system but broadening the view on the complete picture including suppliers, production, maintenance, and decomposition.

SAE J3061 provides an implementation recommendation and guidance for system level and device level picture without limiting and defined specific technologies.

There are many different technical solutions developed and implement. Every of those technical solutions can provide only a single piece in the complete puzzle. Cybersecurity cannot be broke down to a single technology.

Thilo Schumann
CAN in Automation GmbH
Kontumazgarten 3
DE- 90429 Nürnberg
www.can-cia.org

**References**

[1] CANHack toolkit: https://kentindell.github.io/2020/01/20/canhack-toolkit/

[2] iCC 2017: Bernd Elend – Security enhancing CAN trasnceivers

[3] iCC 2017: Olaf Pfeiffer - Scalable CAN security for CAN, CANopen and other protocols

[4] TLS-over-CAN: An Experimental Study of Internet-Grade End-to-End Communication Security for CAN Networks https://doi.org/10.1016/j.ifacol.2018.07.136

[5] ICANN - Root Key Signing Key Ceremony Postponed: https://www.icann.org/news/blog/root-key-signing-key-ceremony-postponed