Chapitre 1

Security in the automotive world

This work was presented in part at the conference of one-legged deaf-mutes in Quiberon in April 1994.

Remoi

1.1 The software revolution

Embedded software is one of the key innovation in the automotive world. From the paper of Robert Charette (Charette 2009) published in IEEE Spectrum, the first car embedding a software was the Oldsmobile Toronado from General Motors in 1977. The Toronado enclosed an Electronic Control Unit (ECU) which managed the spark timing. In 1978, General Motors offered on the Cadillac Seville as an option a trip computer able to display the speed, the fuel level, trip and engine information. This product was based on a embedded version of a Motorola 6802 and had about 50,000 lines of code. Since, more and more functions are performed by software in a car. To limit the volume of cable in a car, all the sensors and on-board unit are now connected to a network backbone using CAN or xxx network. As the computing power of the processor grows, new functions appear in cars. Cars become now on-wheel software platform like airplanes or trains. In a modern family car embeds between 30 and 50 ECUs performing the management of the multiple systems(see 1.1). A premium car can have up to 3,000 singular functions performed by software.

Airbag	ABS	Anti-thief system
Air Conditioning	Speed control	motor management
Turn signals	Headlights	Klaxon
Seat management	Navigation system	Audio system
Wheel pressure	management of the doors and windows	...

Table 1.1 – Software function in a car

In 2009, Alfred Katzenbach, Director of Information Technology Management at Daimler announced that the radio and navigation management system of a S-class Mercedes-Bens contains over 20 millions line of code and the car embeds nearly as many ECUs as an Airbus A380 (excluding the in-flight entertainment system) (Charette 2009). Software in a car has an exponential growth in size and complexity. In just 10 years, the software volume in a car expands by a factor 10, to arrive at around 150 million of lines of code. A model S from Tesla is equipped with a 17' tactical display based on a Linux kernel which controls almost every driver functions. In fact, there is only 2 manual buttons not management by software in the car, the blinker and the glove's compartment.

A side effect of this revolution is the complexity and the richness of the functions proposed to the driver. It is frequent to have the driver manual of a car with over 500 pages to explain all the driver functions. On the other side, we estimate that an average driver uses not more than 20% of those functions. Some automotive manufacturers are asking them the question of a limitation of the software functions proposed in a car. For example, is it really necessary to have the ceiling light of a car going out gradually ?

The growth of the software in a car has also important consequences on the way to maintain and repair a car. An estimation gives that more than 50% of the ECU that are changed by a car mechanic have no software of hardware failure. Mechanic frequently replaces pieces without the knowledge of the root cause of the issue. Now, one of the most frequent activity of a car mechanic is to download and upgrade new version of software. Some car manufacturer, like Tesla proposes to download software updates, including software corrections or new functionalities, using the cellular network without any involvement of a car mechanic.

Aujourd'hui, le coût du logiciel et de l'électronique représente entre 35 à 40% du coût d'une voiture.

Todo list 1 *Introduire la standardisation avec : Automotive Open System Architecture (AUTOSAR). Une solution pour réduire le coût du logiciel en augmentant l'interopérabilité entre les constructeurs. CEI 27272 pour le developpement logiciel*

AUTomotive Open System Architecture (AUTOSAR) was founded in 2003, with the goal to develop an architecture, independent of the underlying ECU hardware that the automotive industry can use to reduce the increasing complexity of software in modern vehicles (AUTOSAR 2003). It is the de facto standard for the automotive software today. AUTOSAR makes an abstract layer of the underlying hardware, so that the applications written on top of AUTOSAR are independent from the actual supplier of the ECU hardware. The AUTOSAR standard documentation guides companies and the automotive industry in designing and implementing software in their vehicles. Besides the Software architecture and the Standardized application interfaces (APIs), AUTOSAR provides a Software development methodology. By adopting the AUTOSAR standard, companies can develop software solutions that are independent of the hardware they are running on, and this software can run on any ECU in the vehicle.

AUTOSAR is a three-layered architecture (AUTOSAR 2000) :

— the *application layer* provided by the software company implementing the specific functions of the ECU. This is the highest layer which contains the software components (SWCs). AUTOSAR application (e.g., ABS, Cruise Control ...) consists of several SWCs, which provide the core functions. An AUTOSAR SWC is an atomic piece of software that cannot be divided and is located a single ECU.
— the *run-time environment (RTE) layer*. The RTE layer provides the standardized interface between the SWCs and the basic software layer. Because of this layer, SWCs can be used on different ECUs, independent of the ECU vendor.
— the *basic software (BSW) layer* that consists of four sub-layers ; the *services layer* providing operating system functions like communication services, memory services, diagnostic services, ... The layer also contains the main security mechanisms. The *ECU abstraction layer* makes higher software layers independent of ECU hardware layout. It provides an application programming interface to devices regardless of their location (internal/external of the microcontroller). The *Microcontroller abstraction layer*. It contains drivers for direct access to the upderlying microcontroller and internal parameters. It makes higher layers independent of the microcontroller. The *complex drivers layer* which provides the ability to integrate special-purpose functions such as drivers for devices that are not specified with the AUTOSAR standard. This layer accesses directly the microcontroller.

Each of the sublayers offers different services as shown in 1.1.

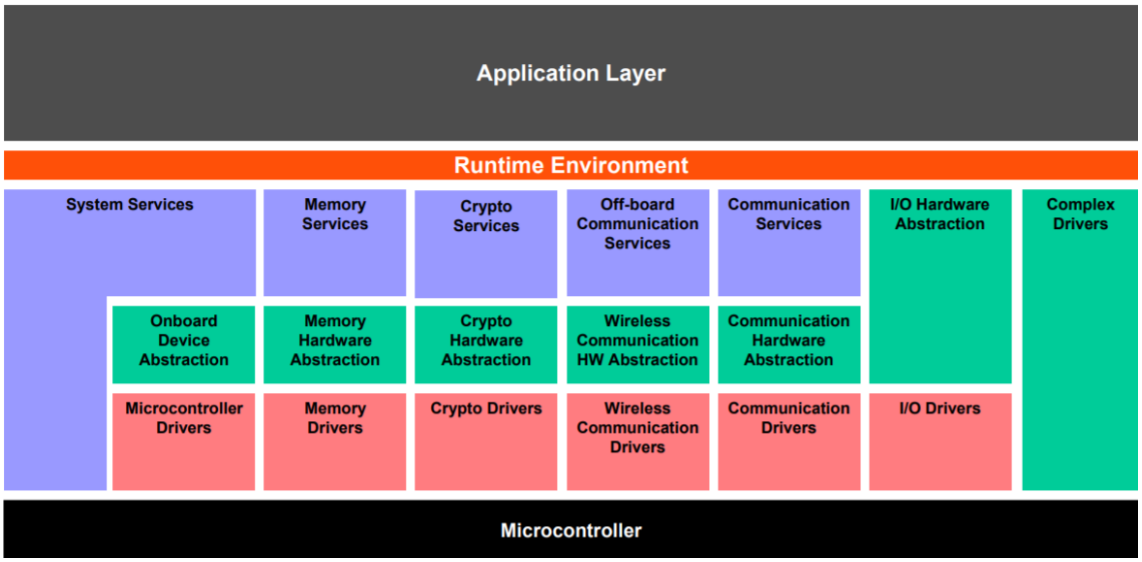

FIGURE 1.1 – AUTOSAR layered architecture

The AUTOSAR standard defines security mechanisms that can be used by the software modules implemented into the vehicle system. It further specifies interfaces and procedures to provide Secure On-Board Communication, and the exact implementation is left for the OEMs to decide on. OEMs choose the cryptographic algorithms and encryption techniques which they want to implement and use in the vehicle system.

Introduction of software in the car permits to have safer and less polluting cars but it has open the door at two new risks the *safety risk* and the *cyber-security threat*.

1.2 Modern car architecture

Todo list 2 *Commencer a présenter les différentes couches logiciel.*

The following four stacks could become the basis for upcoming generations of cars in five to ten years :

— Time-driven stack. *In this domain, the controller is directly connected to a sensor or actuator while the systems have to support hard real-time requirements and low latency times ; resource scheduling is time based. This stack includes systems that reach the highest Automotive Safety Integrity Level classes, such as the classical Automotive Open System Architecture (AUTOSAR) domain.*

— Event- and time-driven stack. *This hybrid stack combines high-performance safety applications, for example, by supporting ADAS and HAD capability. Applications and peripherals are separated by the operating system, while applications are scheduled on a time base. Inside an application, scheduling of resources can be based on time or priority. The operating environment ensures that safety-critical applications run on isolated containers with clear separation from other applications within the car. A current example is adaptive AUTOSAR.*

— Event-driven stack. *This stack centers on the infotainment system, which is not safety critical. The applications are clearly separated from the peripherals, and resources are scheduled using best-effort or event-based scheduling. The stack contains visible and highly used functions that allow the user to interact with the vehicle, such as Android, Automotive Grade Linux, GENIVI, and QNX.*

— Cloud-based (off-board) stack. *The final stack covers and coordinates access to car data and functions from outside the car. The stack is responsible for communication, as well as safety and security checks of applications (authentication), and it establishes a defined car interface, including remote diagnostics.*

https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/rethinking-car-software-and-electronics-architecture

https://www2.deloitte.com/us/en/insights/focus/future-of-mobility/pure-play-so
html

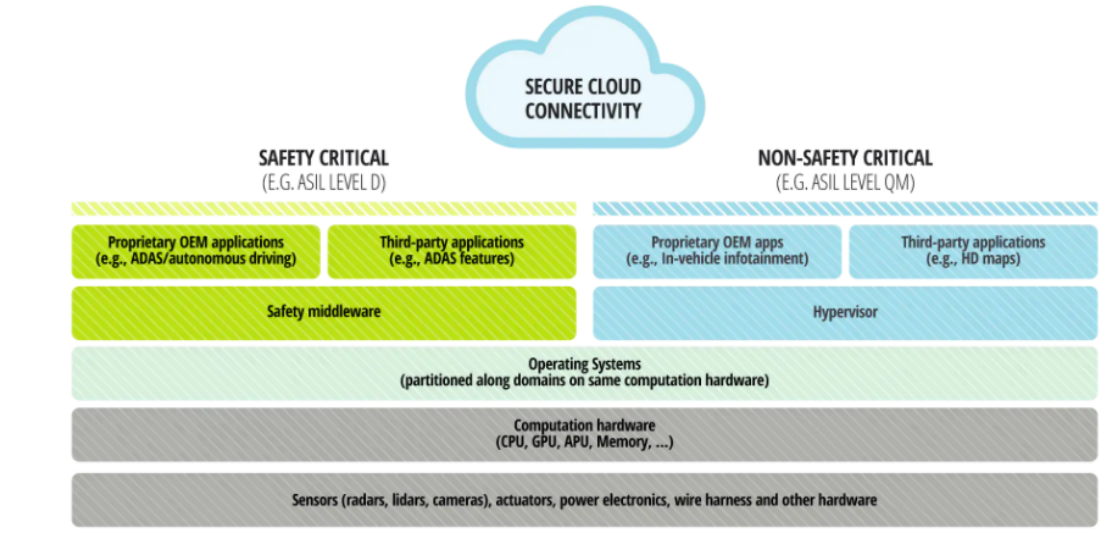

FIGURE 1.2 – Architecture générique

Bla bla bla

1.3 Cyber-security principles

Cyber-security is build on 3 pillars ; the CIA triad.

The security goal is to provide safetymeasures to achieve the confidentiality, integrity, and availability (CIA) triad for protection of the overall system along with its peripherals. The triad CIA is as follows :

— *Confidentiality* : The aim of confidentiality is to protect the critical information from unauthorized users. Confidentiality for network security ensures that the critical assets are accessible only to authorize users.

— *Integrity* : This ensures that unauthorized users do not modify or manipulate the data or information during their network transmission.

— *Availability* : The availability is the last component of the CIA triad that represents the real availability of our information. Authentication methods, channel access, and systems all have to function efficiently to prevent the data and make sure that it is available when required. In short, the availability aims to ensure that data and network resources are available when requested by the authorized users.

FIGURE 1.3 – 3 pillars of the security

Besides the CIA triad, Identification, Authentication, Authorization, Auditing and Accounting (called AAAA) also play an important role for controlling the access to the system resources. The AAAA is a term for controlling the access to the system resources, auditing usage, enforcing policies, and offering the details need to charge for services.

— *Identification* : Identification aims of claiming to be an identity when attempting to access a resource. Providing an identity can involve typing or sending a username or an ID, swiping a smart card, waving a proximity device .... Without an identity the system has no way to correlate an authentication factor to the subject.

— *Authentication* : Authentication is about proving that you are that claimed identity. It requires the subject to provide additional information that correspond to the identity that is claimed. The most common form is to provide a password. Authentication verifies the identity of the subject by comparing one or more factors against the database of valid identities.

— *Authorization* : Authorization is defining the permissions of a resource/object access for a specific identity. It ensures that the access to a resource/object is given the right and privileges assigned to the authenticated identity. If the requested action is allowed, the subject is authorized, instead the subject is denied. It is not because a subject is correctly authenticated that he/she has the right to perform any actions on any resource.

— *Auditing* : Auditing is recording a log of events and activities related to the system, subjects and objects. It purposes to track and record all subject requests and actions. Log files provide an audit trail for re-creating the history of events. It permits to detect malicious actions, system failures but also system performances

— *Accounting* : Accountability aims to reviewing the log files to check for compliance or violation of security policy in order to hold the subject accountable of his/her/its actions. It is also a way of evaluating what services have been used and how many resources have been consumed.

Les principales techniques utilisées pour garantir ces 3 propriétés sont :

— *Chiffrement (Encryption)* : La transformation de l'informations originale (appelé clair) à l'aide d'une clé de chiffrement, de telle sorte que les informations transformées (appelé chiffré) ne puissent être interprétées que par un autre utilisateur ayant connaissance de la clé de déchiffrement (qui peut, dans certains cas, être identique à la clé de chiffrement). Pour être en sécurité, un algorithme de chiffrement doit rendre extrêmement difficile pour quelqu'un de déterminer tout ou partie des informations clairs sans connaissance de la clé de déchiffrement ou faiblesse de l'algorithme de chiffrement.

— *Authentification (Authentication)* : La détermination de l'identité qu'un sujet (personne, logiciel ou équipement). Cette détermination peut être effectuée par plusieurs moyens. Il est généralement basé sur une combinaison de quelque chose que la personne connaît (Something you know) comme un mot de passe ou un code PIN, quelque chose le sujet possède (Something you have) comme une carte à puce contenant des clés secrètes un téléphone pour recevoir un code, un passeport, ou quelque chose la personne est (Something you are) comme une empreinte digitale.

— *Contrôles d'accès (Access Control)* : Ensemble de règles et politiques de sécurité qui limitent l'accès aux informations aux sujets (personnes et/ou systèmes) ayant un *besoin d'en connaitre*. Ce besoin d'en connaitre est déterminé suite à l'authentification correcte du sujet, de par son identité ou rôle. Un ensemble de règles est pré-définie par le gestionnaire de la sécurité informatique en se basant sur la politique de sécurité.

— *Signature (Signature)* : La signature d'une information a deux objectifs ; assurer que l'information signée n'a pas été altérée depuis la signature et authentifier la source de l'information (non-repudiation). La signature consiste en un chiffre dépendant d'une clé secrète connue uniquement par le sujet signant l'information et du contenu du message à signer. Une signature est vérifiable, sans connaitre la clé secrète, par une partie tierce en cas de litige entre parties. Si le contenu du message ou la signature est modifié, la correspondance entre le contenu initial et message et sa signature sera invalide permettant de détecter l'altération et de rejecter l'information. De façon, symétrique, si le contenu du message et sa signature correspondent alors la source de l'information ne pourra pas nier avoir signé l'information car elle est la seule à connaitre la clé secrète.

— *Responsabilité (Accountability)* : La capacité de rendre le sujet responsable de ces actions. Ceci est réalisé en s'appuyant sur des journaux d'audit (audit log). Une fois le sujet correctement authentifié, toutes ces actions sont enregistrées sous forme d'événements dans un journal d'audit. En cas d'investigation ou de façon périodique, le gestionnaire de la sécurité informatique peut réaliser un audit des journaux pour identifier la source potentielle d'une attaque.

— *Sensibilisation à la sécurité (Security awareness)* : La principale source de risque pour une organisation ne provient pas de faiblesse dans la technologie des équipements mais d'actions (ou d'inaction) de la part des utilisateurs du système. Afin de limiter ce risque, il est nécessaire de former les utilisateur

aux diiférents risques informatiques et aux bonnes pratiques pour assurer le niveau de sécurité requis.

— *Sécurité physique (physical security)* : Mise en place de barrières physiques pour limiter l'accès aux ressources sensibles. Ces barriÃ¨res comprennent sont mutiples comme le gardiennage, la vidéo-surveillance, les serrures sur les armoires et les portes, les chambres fortes, l'utilisation de matÃ©riaux insonorisants, ou même la construction d'équipement renforcé (tempest) afin que les signaux électromagnétiques ne puissent pas entrer ou sortir.

— *Tolérance aux fautes (fault tolerance)* : Ensemble de techniques peuvent être utilisées pour garantir le systéme en opération.

	Confidentialité	Integrité	Disponibilité
Chiffrement	✓		
Authentification	✓	✓	
Contrôles d'accès	✓	✓	
Signature		✓	
Responsabilité	✓	✓	
Sensibilisation à la sécurité physique	✓	✓	✓
Sécurité physique	✓	✓	✓
Tolérance aux fautes			✓

Table 1.2 – Principales méthodes de la sécurité informatique

Dans le reste de la thèse, nous concentrerons sur la disponibilité des systèmes embarqués.

1.4 Le risque cybersécurité pour l'automobile

L'ajout de logiciels et de connectivité....
3 niveaux d'attaques :
— *Attaques physiques.*
  — Attaque via la prise diagnostique de la voiture. L'objectif est de pouvoir modifier les caractéristiques de la voiture et/ou de rajouter des options sur la voiture. Les logiciels d'une voiture sont hautement configurables. Un même logiciel est installé sur plusieurs gammes de véhicules d'un constructeur. La différenciation entre les deux modèles s'effectue par le paramtrage du logiciel. Si un attaquant à la possibilité de modifier ces paramètres, il peut activer des fonctions optionnelles de la voiture ou modifier les caractéristiques moteur pour booster le véhicule.
  — Attaque via la prise USB de la voiture. L'objectif est de pouvoir créer un point d'entrée pour une attaque courte ou longue port'ee sur le véhicule. La prise USB peut être connectée sur un équipement radio qui permet d'étendre la portée de l'attaque.

— *Attaque courte portée* L'objectif est de pouvoir prendre le contrôle du véhicule, d'envoyer de fausses informations ou de bloquer les communicatiosn aux

véhilcules aux alentours. De plus en plus de voitures ont un système d'ouverture de portes et de démarrage sans clé.Par exemple, votre voiture est garée devant votre maison et les clés sont sur le petit d'entrée. Un attaquqant peut insérer un équipement radio entre la clé et la voiture (attaque par relais). Un côté est proche de la voiture, lautre côté est connecté à une antenne scannant les fréquences radio de la clé. Ceci permet d'ouvrir et/ou de d'emarrer la voiture même quand les clés du véhicule sont hors de portée de la voiture. Une fois que l'antenne a accroché la fréquence de la clé, elle relaye son signal sur l'équipement proche du véhicule. La voiture a ainsi l'impression que le propriétaire est proche et ouvre les portes et autorise la démarrage. Le véhicule peut être volé sans effraction (voir `https://www.youtube.com/watch?v=_cua7BFX-Qk` pour un video d'attaque relais). Un autre exemple, consiste à brouiller le signal pour la fermeture centralisée d'un véhicule. Le conducteur appuie sur le bouton de fermeture centralisée de sa clé, il a le sentiment que sa voiture est correctement fermée. Mais si le signal est brouillé, la voiture est ouverte et un attaquant peut facilement ouvrir un porte pour voler des affaires dans la voiture.

— *Attaque longue portée.* L'objectif est de prendre le contrôle à distance d'un véhicule. Un des fameux exemple de ce type d'attaque est la prise de contrôle d'une Jeep Cherokee par 2 attaquants ; Andy Greenberg conduisait sa voiture sur l'autoroute vers Saint-Louis, roulant à̀ 70 mph. 2 hackers Charlie Miller et Chris Valasek sont installés dans leur canapé avec leur laptop ouvert. Dans un premier temps, la climatisation de la voiture s'est affol'ee, puis, une image des 2 hackers est apparue sur l'écran de la voiture, la radio a changé de station et le volume a fortement augmenté, Mr Greenberg ne pouvait pas contrôler le volume de la radio ni la station. Ensuite la voiture s'est arrêtée toute seule. (voir Miller et Valasek 2015 pour le détail de l'attaque)