# Secure Over-the-Air Software Updates for Autonomous Vehicle Firmware

By Dr. Wai-Keung Wong

Professor of Computer Science, The Chinese University of Hong Kong (CUHK)

## 1. Introduction

Software-Defined Vehicle (SDV) is one of the next-generation innovation technologies in the automobile industry, which provides capabilities for both in-vehicle and out-vehicle communications. Over-the-air (OTA) firmware update is a need for the SDV firmware update process. OTA transmissions from a software provisioning server to a vehicle telematics module are both over cellular networks and infrastructure/satellites. The firmware images are first sent to the software provisioning server (SPS), and they are multicasted (often over satellite) to vehicle Telematics modules (VTMs), which download them first, validate them and store them in the Firmware/Staging Area. This initial firmware is then flashed to a USB hidden partition in MTD, and then they are moved from there to the Active firmware partition. This determines OTAs delay ratio and also the space for these Partitions on device architecture. At each stage, the integrity validation, transfer, and storage are both prone to various attacks. OTA technology has the distinctive property of running in the full control of a remote over-the-air attacker on all platforms. The described version is severe and similar to the previous two, but a successful attack can now occur by genetic code variation in two steps: over the air and remotely [1].

[2] Vehicles are transitioning to Software-Defined Vehicles (SDVs) with a centralized architecture, leading to increased efforts in implementing over-the-air (OTA) firmware updates through vehicle-to-everything (V2X) technology. Firmware OTA (FOTA) for the electronic control unit (ECU) of a vehicle's drivetrain is crucial for connected cars. Security concerns are addressed through methods like secure FOTA and blockchain integration. However, challenges remain in ensuring complete firmware integrity and reliability, especially in the context of safety-critical automotive OTAs. Consistently, directly downloading and executing core randomized algorithms in-car and injecting random bit

errors, operation conditions, etc., in the case of safety-critical automotive OTAs, can disrupt the proper functioning of vehicles and result in severe consequences, if not handled properly. The challenges to securely maintain the firmware of critical V2X modules of connected vehicles: On Board Module (OBM: telematics and central gateway), cellular Vehicle-to-Everything (CV2X) modules (e.g., eNodeB), Short Range Wireless (SRW) modules (e.g., Wi-Fi), and Satellite Communication (SC) modules. [3]

## 1.1. Background and Significance

It should be brief and not too technical, but it could be rather instructive, more than descriptive, i.e. what the reader is going to learn and why, why he/she should not miss doing so, and why the presentation is interesting in comparison to the clutter of existing literature. In the shorter version, it is important to use an example to introduce assignment alone; in the longer version it is possible to precede with section 1.1 where wishes in mathematics are made. Here, the advantage is to show what intuiting one can reach in mathematics. A proof of completeness of the intuitionistic linear logic embedded in the set theory is presented [3].

The introduction to Section 1 should be able to engage the reader with a promise of solving some existing problems, and also with a vision of the future development of the content, "why the reading material". It should also provide a brief overview on the planned content of the paper. The introduction should clearly present the enthusiastic understanding and motivation of the specific research. It should engage the readers' attention, and including the assignment in the title, the goal should be, when plugging it into a database, to be easy to find if the database is a GTd-search engine for keywords. In this case, "security labels", "optimal Hoare logic", "Hoare triple", "intuitionistic linear logic" and "set theory" could be used as keywords. It is important that the paper reaches the right audience [4].

## 1.2. Research Objectives

Secure over-the-air (OTA) software updates are crucial in the context of autonomous vehicles, providing mechanisms to renew the software of a vehicle's electronic control unit (ECU) through a wireless communication channel [1]. OTA updates offer multiple functionalities, including the ability to improve vehicle performance [3] and to maintain the software. While enabling these functionalities OTA updates can have several edge cases in ensuring the secure update of the vehicle ECU. The system designer will have to consider ensuring real-time properties such as the redundancy of the components and better error control, datapath,

control path splitting methods for update and rollback operations. Consequently, the system designer will have to find the optimal solution for the levels and quantities of the necessary resources, in order to satisfy the real-time requirements respecting the timing consumers and the reconfiguration requirements respecting the requirements of the secure update. to ensure the secure update and verify that all issues and attack are avoided during the automotive field test phase [5]. These crucial safe process thacoupble with some technique that for the moment are denied, since in the test phase all the architects/simulator or the physical gear assuring the safe situation are available.

The main message of this work and its research objective can be summed up as follows. To classify the datasets pertaining to Secure OTA (Over-The-Air) (firmware) Software Updates, proposing the vectorization and k-means clustering approaches. The motivation behind this work is in the necessity to implement secure over-the-air firmware software updates for automotive systems to support the utilities of electric and autonomous vehicles of future years. Future vehicles will have increased dependence on over-the-air software updates to facilitate performance improvements resulting from software enhancements and maintenance of software to address disruptions etc. This is particularly true for autonomous vehicles, which are expected to account for a significant proportion of vehicles in future years.

## 2. Fundamentals of Over-the-Air Updates

maintenance, an embedded system, small code size, software security, software size, RAM.

[6] Over-the-air (OTA) updates, especially secure OTA updates are crucial for protecting software and firmware executing on connected and- in the hoary future, in-vehicle computing platforms in commercial vehicles and modern vehicles. Protecting software in the form of OTA updates is the only reasonable way to prevent attackers from exploiting the vulnerabilities in embedded software. The work aims to establish secure OTA update procedures for cyber-physical systems (CPS) like connected and autonomous vehicles (CAV), which, due to their being connected to external networks, are basically open to cyber-attacks. Coordinated secure and efficient design patterns will be identified and exploited to safeguard automotive updates.[7] Connected and autonomous vehicles, cybersecurity, software update, air interface, network architecture. Autonomous driving features, such as highway autopilot, are the subject of tens of millions of connected and autonomous vehicle (CAV) kilometers in Europe in the last year. Current connected and autonomous vehicles (CAV), equipped with

on-board computers, are initially under-protected, but sufficiently robust against cyber-attacks. However, these days the 25-year-old automotive ECUs are being replaced by more than a dozen of automotive ECUs [1, 2 finds all contributors quickly]. Another problem is that most ECUs were developed by large automotive companies, not by independent embedded software providers. Restrictions are imposed today to perform over-the-air updates and software installations in a modern vehicle to provide cybersecurity automotive systems [4, 5] protection against malware and also to save development, testing, deployment, maintenance and as well as software product life cycle management. Recently proposed secure OTA update technique explicitly considers the usage of network-based software update infrastructure on the server side. To provide security in a network-based software update system, the frequent practice is prefixing software updates with integrity checks, signature verifications, encryption. Both symmetric (AESX) and asymmetric (RSA2048) cryptography, protocol communications

## 2.1. Definition and Overview

Additionally, the software update process should continue at the OTASP by maintaining confidentiality, safety, integrity with stringent bounds on additional communications and delays [6]. This paper aims to present a novel secure SD. We provide a reference to the safety and security components, guidelines, and analysis for this new stack, but we also encourage the reader to perform the security analysis for a new stack before deployment on an autonomous vehicle. Additionally, the results can be handy to define safety and security guidelines when deploying a secure connected vehicle software update system.

[3]Updating automotive Electronic Control Unit (ECU) software over the air (OTA) has become possible with the recent evolution of wireless communication technologies and architectures, which is significantly essential for the safe operation of connected and autonomous vehicles. However, to guarantee the success of OTA software update distribution, we need to ensure the final updated ECUs' safety, integrity, and confidentiality against passive eavesdropping and active attacks [1]. In that sense, we need to design secure protocols and algorithms for the ECUs to accept OTA software from the update server and perform efficient and secure software installation.

### 2.2. Benefits and Challenges

The vehicle firmware is the foundation upon which the operation and safety of the vehicle is built and it is, for these reasons, essential to protect this layer from miscreants with malicious intent by building in security measures.rema Remainder of article organized as follows. Improve the OTA protocol by assuring the authenticity, ensuring confidentiality, encoding the integrity of update packages, periodic checking and compatibility checking, unwiess data-type checks are proposed in the third section.

The State of Technical Service (SVV, 2016) and annually more than 2 million vehicles pass the Swedish Periodic Technical Inspection (PTI) where they claim each car owner spends 4,495 Swedish krona every month to service car. That adds up to 54 billion Swedish krona spent by the car owners per year on car service. Therefore it is important to prevent the attacks through an efficient and secure update process. [2] Extensive accessibility of the electronic components in a vehicle and the possibility of OTA from update sources located outside the vehicle represent considerable challenges in vehicle security. A successful attack on these devices can cause potentially fatal consequences that affect directly the safety of people´s life. For this reason, it is important to defend potential threats through a secure updating process. OTA updates are excellent vectors for the compromise of the vehicle firmware.

[1] The first rational reason for deploying OTA is due to cost efficiency. Vehicle producers have to bear substantial expenses during car recalls. Nearly 800,000 vehicles have been recalled in Sweden only over an eight-year period under 200 to 400 Swedish krona per vehicle (Swedish National Road and Transport Research Institute, 2020). This makes a sum of 100 to 320 million Swedish krona. The majority expanse is spent on labour, that is according to Aftonbladet (2014) in Sweden 174 billion Swedish krona is spent per year for car repairs.

## 3. Security Considerations in Autonomous Vehicles

In a particular aspect, a total of eight communication services are defined and described to support the adaptation of the services. To secure these communication services, the security services secure the data in these communication services. These services are protocol independent and can therefore not be used only for E/E-architectures as E/E-architectures are described. One of the realized applications of the method is to secure the whole Tesla onboard power supply ECU. In general, the security services are distributed to the participants of a vehicle. For instance, for steering, braking and electrical propulsion, the

corresponding ECUs have to carry out their own security service, which are described within this safety and security concept. All ECUs with strong functional safety and security services like power supply and steering, need a backup ECU to guarantee the fulfillable security and safety level. No extra cloud or external backend is involved, which is like 5% of the realizations inside modern vehicles and is called here "specialized solutions." Two vehicle software architecture will be considered in this article.

[1] The security of software update mechanisms is crucial for modern automotive systems. The common practice combines the existing internet access of a vehicle with an encryption schema, digital signatures, and updates by public keys to ensure authenticity and integrity during an over-the-air (OTA) process. Some systems, as exemplarily discussed [in 5], even manage the update process themselves without any additional backend system [8], [2]. Problems can arise on top of the architecture of the update process and its authentication mechanisms, especially if it is implemented on electronic control units (ECUs) with real-time requirements and varying update rates. With the growing complexity, also the comfort functions and standalone embedded control loops require more project-specific safety and security concepts. A central aspect of a modern vehicle is the communication and the networking of different ECUs on different hardware and software levels, each with different real-time requirements. A quasi-real-time capable network has to ensure deterministic message transmission times, like TTEthernet, Time-triggered Ethernet, as well as stringent security requirements, whereby timing constraints (e.g., bandwidth guarantee) and different risk classes (e.g., unplanned restarts) of integrated ECUs have to be considered. With estimable computation power and communication protocols, the combination of computation- and communication intensive safety and security services for a remote management of vehicles (coordination of update processes, synchronization, failure recovery, etc.) improves the user comfort as well as the vehicle uptime. A new feature of especially modern vehicle descriptions is the existence of backward-forward interactions, in which a vehicle interacts with the outer environment and reactively adapt its internal actions to the environment state. The communication between software components within individual ECUs, as well as the external communication from these ECUs, is also specified in the vehicle description.

### 3.1. Vulnerabilities and Threats

Cybersecurity strategy, including security considerations handled when the vehicles are developed, has evolved in car OEMs. Moreover, vehicles are equipped with electronic control units (ECUs) connected together through buses, which include the in-vehicle, local, and WAN segments. Communications in the WAN, of course, refer to traffic through internet-related services. Furthermore, connected cars are highly dependent on software. Their lifecycle includes development phases, when they are designed and created at OEMs, quality-assuring phases that are performed at suppliers, right up until they are delivered to their final customers. When modern vehicles are fully connected, it is unsurprising that they contain telematics devices that are in constant communication with external networks. The links between vehicles and external circuits form quite a wide area network. Therefore we can say the vehicle can often be identified as a system on a chip(SOC).

[9] The increasing number of electronic systems and connectivity in cars, along with the rise of fully connected and autonomous vehicles, has a direct impact on mobility. Vehicles could once transport just their drivers and passengers, but there is now a need to securely carry data back and forth between their various parts. The automobile is therefore no longer a mere vehicle, but also a day-to-day life activator that connects with the outside world in numerous ways. As a result, vehicles have become technologically complicated and now contain many different electronic control units (ECUs) connected together through multiple buses and become a platform that forms a cyber-physical system. The connected vehicle ecosystem, as well as its communication structure, has already been introduced in. Threats to security manifest here in potentially damaging ways, much like other mobile objects or data networks. It is unlikely that these threats would be discovered after a security incident, such as hacking, has occurred.

### 3.2. Secure Communication Protocols

In this regard, data protection mechanisms like the Hash Function (HF) and public key cryptography is needed. As asymmetric cryptography is a very computation intensive process, at least when considering key exchange and more importantly decryption, in contrast, symmetric key encryption promises to provide an efficient technique once the keys are exchanged, the main question is how the symmetric keys could be exchanged [4]. In a connected vehicles scenarios, the required symmetric key may either be exchanged with every

single vehicle, a huge challenge. In an ideal scenario, the identical key is used by all vehicles and would only have to be exchanged with the manufacturer.

Secure communication protocols like the Transport Layer Security (TLS), in the context of the Internet, and DTLS, which is used in connection with the Internet of Things, are usually employed to guarantee an encrypted communication of two peer devices [10]. TLS and DTLS can be used successfully in an appropriate automotive environment. Nevertheless, it should be noted that both protocols are associated with substantial overhead when using them in an IoT context. Furthermore, TLS/DLTS itself cannot guarantee the integrity of the transported data (firmware) nor can it avoid that the same data was manipulated by an attacker.

## 4. Existing Solutions for Secure OTA Updates

Securing over-the-air (OTA) software updating for application software in the vehicle is necessary given the potential new attacks that could exploit the software update feature. The following software security mechanisms can be applied to deal with attacks exploiting the software update feature: RSA encryption key security, two-way SSL server authentication, hash function one-time password (HOTP) for web server authentication, hash-based message authentication code server and client authentication, a blockchain-based RSA encryption and security checks. These technical measures are more or less feasible, but cryptographic operations are not suitable for high-level automotive devices where performance is required and other operations are simultaneously important. Another problem is that ECUs of AI vehicles might experience cryptographic operations-related attacks as proposed by Bellare et al. Like in, solves a similar problem of hardware-related attacks and performance degradation with a secure technique involving linked keys, an encryption of software packets, and a random scheduling in order to detect unexpected behavior in key. After successful security checks, Liang et al. in calculate the orbital optimal maneuver by using meta-heuristic algorithms where the spacecraft keeps the closest position possible to the nominal path, predicts the future position and the optimal move to return to this signed point. In this work, we want to propose an update strategy to generate and securely apply the minimal amount of OTA updates justifying and investigating, through the help of zero-knowledge proofs, the presence of updated security issues. This work also fits within the general concern of autonomous vehicles of providing innovative performance for all electronics, telemetry, and synthetic probe sensors in addition to physical tological and physical phenomena heuristic.

The uniqueness of this work is the fast secure OTA between vehicle and the cloud, guarantee of privacy, and the working time reserved by the vehicle out of cabin. This paper is organized as following.

[4] [2]The increased complexity and modularity of the Electronic Control Units (ECUs), the introduction of new features, and the forthcoming transition of vehicles to 5G makes the support of automated and secure over-the-air (OTA) updates for the vehicle firmware necessary. Current commercial OTA solutions come with many limitations or drawbacks like no support for real-time scheduling, heavy runtime overhead, no support for the privacy policy, and inability to minimize user involvement. The emergent secure OTA solutions offering privacy and efficiency of updates are mainly based on autonomous vlans, no known location, and network-processing based vlans. However, these do not directly support the modular and fine-grained separation of the application software among the different ECUs of the vehicle. Thirdly communication security among the different ECUs is ensured by the secure channel set up by means of the LSB trends, no interest IM signals, and no override IM signals. The current security and efficiency of the OTA solutions are limited by human impact, instantiation, and intrusion [10].

### 4.1. Key Technologies and Approaches

There have been several research works done in the area of Security upgrades of connected cars. However, most of these have concentrated more on the security issues relevant during the upgrade process rather than the performance and reliability of the updates repair process especially in the domain of Software Defined Vehicles. Research has also been done on getting a clear understanding of how these upgrades are made on the firmware. These deployment methods commonly use HTTP or HTTPS, however, have the advantage of being tried and tested but they use these protocols which have high overhead and hence are not compatible. [ref: 758349a9-097f-4a4b-892d-abd65a81ef0b, 74120af5-2bd7-47be-bb1f-373c76765f10]

Symmetric key encryption largely relies on the OE key which is susceptible to attack. The hash function only provides integrity while the efficiency and fault tolerance of the blockchain for V2V, V2X communication are issues in their adoption for the given domain. The RSA restricted to integrity only. They propose steganographic hidden key embedding as a simple technique. However, they do not consider the reliability, scalability, network and implementation related issues. Also the attack surface can be still be large.

### 4.2. Case Studies

All these reviewed articles have discussed the different views of secure OTA updates in connected cars. They have proven the importance of secure OTA updates in connected cars and have been able to support the claim with plenty of evidences. Literature gives the possibility to the reader to obtain a complete view of the current situation, but the a complete study of the state-of-the-art shows clearly that no single method is efficient by itself, and even those methods that meet well some fields often lack in others. The definitive method to the improvement of OTAs should consist in the combination of different algorithms in a suitable way, if not consider new solutions, without eliminating current methods [6].

The five reviewed articles are able to represent fully the research paper outlining the most innovative and current security advances in Over-the-Air (OTA) software updates for modern and connected vehicles: it can be divided into three main parts. For the UDP-based UpdateQOTA, the data transmission is less secure. Until now, the communication between the vehicle and infrastructure is still vulnerable, which enables an adversary transmitter to inject false message into the queue as a Denial-of-Service (DoS) attack. The proposed D2Qtree can dynamically maintain the authentication of each message at the transmitter and receiver, which is hard to be cracked by current adversaries [4]. The blockchain technology has been proposed as a means of handling security limitations, especially in queuing strategies and the data storing and accessing capabilities. However, the processing latency of block-î€‡chains always increases with an increasing number of blocks, so the relevance of earlier blockchains diminishes.

## 5. Proposed Framework for Secure OTA Updates in Autonomous Vehicles

In [5], the authors discuss the importance of secure OTA firmware updates. They focus on the pre-application phase of secure OTA updates, where games among the vehicles, update server, and OEM may start when the OEM releases the update, and the vehicles that inherit these updates initiate the corresponding turns respectively. The game is originally called security competition game (SCG), which can be a possible usage scenario of blockchain in decentralized attribute-based encryption (dABE). The attribute combination can change constantly, due to the complex network conditions in the vehicular ad hoc network (VANET). A prominent feature of this mechanism is that enormous computation burden can be offloaded to the edge server rather than the vehicles, and dABE permits the computation

outsourcing as e.g., for traditional ABE, distributed attribute management is necessary. However, if all vehicles in an ongoing challenge game are to be processed by the edge server, it will become a bottleneck.

In [1], software over-the-air (OTA) updates are highlighted as a critical aspect for autonomous vehicles to provide new functionalities, bug fixes, cost-efficient service, and optimized vehicle performance. As a widely-used practice, infrastructure and ecosystem for OTA software updates have been very well established in various applications and operating systems. Despite IoT devices, consumer electronics, and other industries taking advantage of OTA updates due to their efficiencies and convenience, different factors such as security, communication medium, and environment, are going to alter the ways in which software updates are performed. OTA updates are crucial for user authentication and security, as well as for fine-tuning systems in IoT devices and consumer applications. However, it is very challenging to apply OTA software updates in mission-critical systems (e.g., safety-critical autonomous vehicles), due to several important properties, including potential unauthorized ECU access, potential cascading hardware failure, safety of passengers, safety of the surroundings, and tamper-resistance for feature patenting. These short-term liabilities require effective and adaptable strategies. OTA software updates are crucial for defensive cyber security in vehicles, but they also increase the attack surface exponentially due to their critical, gateway-like as well as tree-based structure.

### 5.1. Architecture Design

To meet these challenges, the architecture of the secure FOTA system is designed and organized as shown below: According to modern vehicle architectures, an advanced secure FOTA system has been designed and modularized to ensure efficient setup and execution in different types of FOTA systems such as OR,OP-V2X, IV, fog and edge… The designed secure, architecture of the FOTA system under-doubles four levels of micro, support, Edge, and Cloud and dual sufficient sub-architectures, such as computation units that work to provide direct physical hardware facing, physical facing of the service delivery, data privacy and secure. While the Aumann–Morron & Hetzer (AMH) possess clear functions as a physical interface, a usual server, database and analytics servers [7]. Through the organizations of several components in different layers and sub-architectures, it is possible to quickly, easily, efficiently and securely implement secure FOTA systems required for V2X connectivity, Driver Devices, Foga Edge clouds.

Secure Firmware Over The Air (FOTA) plays a key role in realizing vehicle connectivity and autonomous driving systems, as FOTA updates can continuously enhance vehicle performance, security and driver experience, and fix hardware and software faults over the air. However, it is challenging to design secure, efficient, modular, and universal FOTA systems, as compared to conventional FOTA systems, vehicles are equipped with newer electronic control units (ECUs) higher network connectivity, distributed architecture having Edge and Cloud computing and built-in vehicle dealing with environmental constraints [2]. These peculiarities may require different FOTA solutions, such as specific foreauthentication and on-line software updates, to counter security, integrity, reliability and containment of all types of adversarial attacks; difference in the total time required for FOTA execution; different Efforts to be taken by the OEM for designing and integrating the FOTA systems in vehicles. The centralized configuration for FOTA will not work properly on upcoming vehicles due to heavy computing and communication constraints in in-vehicle networks or controllers, and small free airtime for software updates, as compared to a cloud [4].

### 5.2. Key Components and Functions

The main components of the system are three core functions including the Reporting and Monitoring System (RMS), the Security Gateway (SG), and the Vehicle Communication Unit (VCU) [2]. The core functions of the Cloud Backend would be Register Vehicle and PP generation, Forward Requests to the SG, Manage Customer List, and Authenticate and register the sensors. As interface to the above described components a new requirement on the Cloud Backend must filter the reporting of all registered sensors against the list of registered vehicle fleet to verify if all Sent Reports are related to a vehicle or should be discarded [6]. The RMS component of the system provides Representation of the communication of the individual reasons of the VCU to the SG component, away from management of the procedures for the collection of reporting requirements, to the coordination and execution of all communication logical data flow to the SG, to Start and Supply into GCAP-enabler of PARs, to Manage the parameters for the VCU and its sensors, Handle and Manage Faults and Report Reports, Manage responses flow from Mlb to VCU/VSU, to Report the Probes of serverListProbes flow as the result of one forwarding from SG.

**6. Implementation and Testing**

An OBD2 connected electronics device (e.g., smartphone) is the customer-facing element for all OEMs and aftermarket service providers. It is the central device complemented by OBD2 sensors that monitor the tire condition and vehicle-location information that supports sensor-cum-actuator systems. The communication speed of OBD2-connected devices is another important requirement. At 100 kb/s, the OBD2 communication standard data rate is compatible with the new standards Bluetooth Low Energy (BLE) 5.0. Therefore, from an implementation complexity, compactness, and rapid development-time context, BLE 5.0 is the most optimal communication channel [1]. A secure, pseudorandom one-time password that is valid only up to the key link creation of the BLE device is used as a cryptographic primitive. Finally, a Bluetooth security feature known as BLE Secure Connections provides a combination of key management and payload encryption and decryption mechanisms, ensuring that secure communication is maintained without eliminating the real-time constraints of OTA updates.

The prototype employs the MQTT and Merkle tree technologies to create a FOTA protocol, called "MQTree" [2], to provide access control and ensure data privacy and data integrity for secure software update over vehicles connected to the internet. In the embedded platform, the vehicle sends encrypted authentication values received from the temporal key together with the new encrypted firmware in a secure manner to avoid the replay attack and the parallel OTA attack. Additionally, OTA firmware and software update are needed not only for security reasons but also to handle various problems related to vehicle software. Secure OTA update solutions also need to address consumer trust and privacy fears. We are in an era where, for every product that is developed to meet a specific need, an electronic system with sensors and actuators (Internet of Things (IoT)) is developed in parallel. For this reason, IoT devices not only need to keep pace with the development of standards to improve user accessibility and usability, but also need to be developed with suitable security infrastructures [5]. The Multi Layer Cyber Security (MLCS) approach offers detailed guidelines for various security mechanisms that should be adopted when designing and developing a secure IoT network supporting Vehicular Connected Physical Objects (VCPOs). This paper describes the integration of a very modern communication channel with electronic devices' embedded secure software update platforms to overcome these challenges in terms of term security, performance, ease of use and cost effectiveness.

### 6.1. Simulation Environments

When an OTA software update is proposed, OSCR validates the authenticity of the updates using the property in theorem 4. The theorem is proven if the cryptographic engine is well implemented and function as designed. However, it requires human expertise to prove the general case. In this work, we aim to enhance the variability and identity in testing the system without any human intervention. Hence, the second line of the simulation does not include a model of the cryptographic engine module. We prove that if the sending and receiving of different variations of the parameters and the model of the cryptographic engine are found to be incorrect, then the cryptographic engine is significantly likely to be damaged [4]. As future work, proof of this verification has to be performed for the actual OSCR implementation in an embedded system.

A crucial part of our proposed OSCR is its testing and evaluation. The logic and verification of the boot loader and communication protocol are primarily validated. In addition, we present and discuss the choices for a set of simulation environments closely matched to the requirements of our case studies [11]. We make the simulation environments publicly available (see Datasets) such that future work that aims to contribute to the validation and/or extension of OSCR can extensively evaluate their system. We presented a system called OSCR which stands for Over-the-air Software releases for Control computer Realisation.

### 6.2. Security Testing and Validation

This chapter addresses the problem of secure (over-the-air) firmware updates for autonomous driving platforms, where updates have to be carried out to ensure the overall security and safety of the platform and other stakeholders: ECU customers, component and system suppliers, as well as the end-user—the autonomous driving system user. One representative is the approach which integrates public key infrastructure (PKI) and attribute-based encryption (ABE) with digital watermarking [5]. The software over-the-air update (SWOTA) protocol in a consistent framework that encompasses message flow, attribute-based encryption, and watermarking schemes for secure SWOTA in an attribute-clustering based automotive embedded platform. Secure software over the air updates are essential for safety and security reasons in the case of autonomous vehicles. The continuous use of outdated software systems in any vehicle or vehicle component could lead to exploitation of unpatched security vulnerabilities and hence potential attacks on the overall system. This is particularly undesirable in the domain of autonomous vehicles. There are numerous works related to

secure over-the-air updates. Yet, most of them focus on limited aspects of the secure update process and lacks a complete focus on the security implications of the various steps in the update process. A comprehensive test and evaluation (T&E) approach combines penetration testing and traditional SWOTA testing at the OEM level only to consider facets of secure software update. The test and validation aspects mentioned above mainly focused on preventing unauthorized access to the vehicle and to its components [12]. Key management is an essential aspect of a secure software update process and hence its testing and evaluation is crucial. Reference [7] identifies this problem.

Common testing techniques for potential insecurely-implemented FW and the offering a systematic and structured way of conducting security validation as part of the test campaign for the implementation of an over-the-air firmware update process. The proposed security validation approach can be applied for remote and on-the-fly (secure) update processes in a vehicle, and is an extension and specialization of widely used security validation strategies for ICT systems to take unique aspects of automotive environments and use cases into account. In contrast to general platform-level security testing of common ICT systems, automotive platforms must additionally fulfill safety and security regulations and norms.

**7. Performance Evaluation**

This work aims to empower OEMs to update the complete firmware of the ECUs of their automotive vehicles by harmonizing the E&EtE-embedded security mechanisms for securing software life-cycle, highlighting the main operation processes and not yet addressed aspects to implement g-ECU software updates [2]. In this respect we propose the design of a Secure Unified g-ECU Update Manager, which is a cloud-based symmetric Crypto-Scheme for encryption/decryption and a cloud agents distributed software structure. The Crypto-Scheme handles the secure g-ECU update process while the cloud agents schedule the g-ECU updates and interact with the Crypto-Scheme facilitating block signatures. We present as security solution provider the secure update sub-layer, a part of the Cloud-Edge solution Secure g-ECU update platform (SecUp). The SecUp nodes are performance evaluated and benchmarked for several g-ECU types.

Security and safety are paramount for autonomous driving [3]. Integrity, authenticity, and confidentiality of software running in an electric control unit (ECU) must be guaranteed to ensure security and safety. Over-the-air (OTA) firmware updates for autonomous vehicles are

presented as a promising solution to improve efficiency, update-ability, and even security. Most autonomous driving applications and standards are moving towards a remote maintenance mechanism with respect to software lifecycle in software-defined connected vehicles, called, OTA firmware updates. This calls for new requirements to update the entire ECU firmware remotely, which is called the global ECU (g-ECU) updates.

### 7.1. Metrics and Benchmarks

Consequently, we believe that the security of software updates and the secure software updates problem in connected and autonomous transport systems have not been adequately studies. In this paper, we provide a general description of secure software updates for urban transportation and autonomous transport systems, introducing a hierarchical approach to solving these security problems. The hierarchical approach we adopt allows us to create three abstract models, which we then apply to three tailored scenarios we believe are representative of the connected vehicle sub-domain (software updates for in-vehicle ECUs, traffic light controllers and road side unit software configurations). We focus on the concept of Privacy-Preserving and trust for Software Update and summarise the common discussion around the OTASU both in an abstract fashion and in realistic scenarios. For the model samples, we present known work on establishing secure software update delivery mechanism using state-of-the-art techniques like Attribute-based encryption (ABE) [13].

Cyber security for automotive systems has been discussed for some time now. Different methods and concepts demonstrate the potential for secure designs and implementations in the automotive sector [1]. The proposed models are usually evaluated empirically and most notably with test bed implementations; showing quite high security properties. Using benchmarks to allow for an empirical comparison would also provide a possible future research direction while this approach will offer a more formal comparison with models according to their security attributes(table 1). Metrics are used exclusively in this field of study and albeit they are good at providing a qualitative/quantitative view of the attributes, but they can be subjective of the case study implementation.

### 7.2. Results and Analysis

The firmware updates are responsible for using exponential autoregressive technologies in the bandwidth allocation. In order to provide instantaneous convergence to the required level of representation error, the cross-characteristic module may resort to: 1) Crisscrossing Zed-

Bull remapping; 2) T $h$ = 0 and T c = 0; 3) p = 0 [11, 2]. The need for an efficient U ∨ -constellation implementation is discussed in. The driver can reach their destination in favor of his escort capabilities [2], while the journey may involve non-negligible special relativity characteristics at the corresponding time frame (t = 110 ms for the post-SeZ). Compatible physical layer communications require management of selective relays, transmitters and adjacencies, from the initial stage of intra-car (palmasculo) versus inter-car relaying points. Random fields represent the sites corresponding to an enormous broadcast pre-scattering and Delaunay–Voronoi pattern clustering. In the meantime, we have established a global length scaling, resolved weak but non-negligible Euclidean elements, and obtained probabilistic applications by maximizing internal timing nodes in relation to the global network of nodes.

When the theoretical alternatives for a cyber secure communication network are considered, a number of security vulnerabilities associated with the modern network operation come into view [10]. These security vulnerabilities are distinct to such an extent that some of the attacks cannot be easily detected, whereas some are generally benign or yield transitory threats to the digital mail channel. However, a secure communication channel is needed to support the finances on these time-sensitive networks. The vulnerabilities can be broken down into three classes: (a) Attacks leading to complete immobilization of vehicle software; (b) Attacks with minimal energy consumption and extended operational time; and (c) Attacks making vehicle software vulnerable. The communications channel can be CVE-2018-11777 to render the mail server software completely unusable by injecting spam into the system. However, it is challenging to even detect these cyber attacks that can be catalyzed by the firmware updates released by modern vehicles.

## 8. Future Directions and Research Opportunities

[3] In this research, we identified the most popular secure FOTA methods, which are over-the-air architecture (like-OBD2, connected car model, direct update), used communication protocol (like MQTT, HTTPS, CoAP), and secure method of FOTA (like signing, certificate, encryption). While mentioning FOTA methods, emphasis was especially on secure ones. Furthermore, we discussed this update and publication through various books, master's dissertations, scientific research study articles, journals, open format papers, conferences and symposium publications.[2] In along with UDP-based protocols, the security weaknesses in MQTT-based FOTA are introduced at the data link layer of the OSI network architecture. A

software-defined car integrates various types of software modules into a single piece of hardware and uses the new architecture CIS (Centralized Integrated Systems) with V2X technology. To push this architecture to the vehicles, secure OTA software updates were proposed. Practical approach for traffic shaping in Data-Oriented Communication (DOC) was introduced in. An approach to secure in-vehicle networks from attackers was proposed in, and this approach was evaluated in a simulation in. Additionally, a false peak in the frequency domain could be used to attack the filter structure directly in. Comparing the most popular protocols used for secure MQTT communication and their security weaknesses in showed that a secure protocol needs higher computational power. None of the protocols could be considers to be a panacea. Mutual extensions in data prototyping languages most noticeably the Turbo Extensible Binary Languge (TXBL) were examined in and an increasing exchange of messages over the internet was confirmed in an interesting way in. Here, the detailed description of the functionality and command possibility of a chat bot, as well as the Twitch and Discord web hooks were given. No direct indications for car use can be derived from the posts.

### 8.1. Emerging Technologies

The severity of the adversarial attacks forces us to consider emerging technologies for secure communication and secure OTA software updates. The presented blockchain-based secure OTA protocol will help to achieve mutual trust and integrity between vehicle and server. The trust between the vehicle and the server is achieved by using a public key infrastructure with different asymmetric encryption schemes for vehicle-server communication. A novel hybrid application of a blockchain protocol is used to ensure the integrity of over-the-air software update payloads that are provided to vehicles. Initially envisioned to manage only blockchain databases for cryptocurrency trading, the consensus algorithm (incentives of reachability) and chain of blocks technologies are being considered for more complex services in different promising fields in e-health, energy and automobile networking. The proposed model is not only capable of supporting FOTA systems; it also supports any other kind of communication taking place in the vehicle drivetrain derived from MQTT communication, such as smart mobility applications, intelligent transportation services (ITSs), vehicle-to-infrastructure (V2I) communication, vehicular platooning and vehicle-to-everything (V2X) communication.

To remain competitive and without losing market interest, new software releases will need to promptly address critical software bugs and introduce new features or functionalities [14].

The heavily increasing level of automation, combined with high connectivity support will gradually require new paradigms in the way we communicate with our vehicles in order to maintain and update software on the electronic control units (ECUs) and advanced driver's assistance systems (ADAS) [2].

### 8.2. Regulatory and Standardization Efforts

The 5G Automotive Association (5G AA), a cross-industry members' organization of companies from the communications and automotive sectors, is working on creating a V2X communication standard. The first part contains information for the basic communication security services, e.g., authentication, integrity protection, and privacy. Moreover, the document suggests rolling out a European Union and a UNECE Indonesia Level 3 automated-driving policy in part of which a vehicle telematics service needs to ensure security principles for identifying a hazard on the left–right lane including rare edge cases like a VI [5].

[14] [2] Currently, different organizations, forums, and alliances are working on regulatory and standardization efforts to enforce and promote the secure usage of OTA update services. It should be noted that the existing regulatory frameworks such as the General Data Protection Regulation (GDPR), UN-ECE "R" series, Road Transportation Auto Cyber Security - Cyber Activities (RTA-CA) and UNECE World Forum for Harmonization of Vehicle Regulations (WP.29) Regulations (R155) are not specifically addressing the OTA update security requirements . The following section outlines the current state of regulations and standardization efforts that involve vehicle telematics services, TCU, and connectivity with the backend systems for the developed secure system. In the following section, till the most recent standards we could find are IEEE 1609.2a-202x and IETF Internet Standards Track Documents. In the following, we outline the implication of the aforementioned standards in the AV domain. The main challenge for the existing regulations for the V2V and IVN communication protocol is the retrofitting period with fully AV-dominated fleets in the future.

## 9. Conclusion and Recommendations

Research efforts in the field of in-vehicle communication security has attracted the attention of researchers and practitioners alike. A move towards the use of electric vehicle and autonomous vehicle technologies have sparked industry-wide initiatives to secure the in-vehicle network, which includes the inter-vehicle communication, vehicle-infrastructure

communication, vehicle-cloud communication and OTA software update services. In this article, illustrated the security standard for integrating wireless communication technologies with electric vehicles/autonomous vehicles. Thus, it is important to address security challenges that arise in different communication instances involving electric and autonomous vehicles. Nowadays electric vehicles, driver error-free self-driving technology integrated autonomous vehicles are frequently discussed research areas. Due to the revolution in the use of electric vehicles and autonomous vehicles, providing safe and secure communication technology has become critical. Autonomous vehicles (AV) are equipped with sophisticated technologies, including onboard sensors and wireless communication protocols for perception and data sharing with other vehicles, traffic management systems and other vehicles, i.e., V2V, V2I, I2V etc. Similarly, as in electric vehicles, the batteries of the autonomous vehicle also come with challenges, including how the batteries can be used efficiently and how their health and safety can be maintained. In this context, the intelligent in-vehicle battery management system (BMS) which is designed to optimize the use of the battery and to monitor its key parameters ensuring safe, reliable, and efficient way of operation is considered. The diverse communication needs and the performance requirements of the sensor equipped, electrically powered AVs are integrated into automotive data communication and computer networks. The networks are engineered to be used by electrically powered AVs, to enable the communication of the charging station and electric vehicles. However, the safety of autonomous and electric vehicles is paramount and must not be overlooked, which is of prime importance for us to design and develop the vehicle systems to adhere to Ultra-Safe automobile standards.

[5] [10] [7]In this work, we proposed a security mechanism to provide secure over-the-air (OTA) software updates in autonomous vehicle firmware. The proposed mechanism provides all the salient security properties that a secure OTA updating process must provide. These include: (1) Confidentiality, to protect the firmware of the vehicle from unauthorized access; (2) Integrity, to ensure that the update file has not been tampered; and, (3) Authenticity, to ensure that the received update genuinely comes from the manufacturer and is not fake. We demonstrated that our scheme works under various realistic security analysis assumptions.

## 9.1. Summary of Findings

9.1.1 In this chapter, different mechanisms have been proposed using linear logic and known to have potential for various new applications. POPULOT is an IP to Token vending library

and was initially thought to be used for establishing VPNs securely. POPULOT is the first step towards secure VPN setups with or without SDN. Similarly, ILSTT generates a token instead of converting into an IP address, which again is a step towards having VPN services automatically by getting a token. The authors have also used ILSTT to securely distribute the secret keys. This concept is beyond SDEE as such either there has to be a direct link or the egress path must be a tunnel (IP-in-IP by design) over Internet.

Manufacturers have identified a cost-effective way to fix vehicle flaws without recalling vehicles through the use of OTA (Over-The-Air) updates. However, while OTA updates have made life easier for the customers, they also bring in risks. Updating firmware (or software) is a double-edged sword which can either fix some CVEs/bugs or introduce a new CVE in the current ecosystem of secure autonomous vehicles as discussed earlier [10]. [6]. The CAV ecosystem is moving towards Internet-scale Software Defined Networking (SDN) [4].

### 9.2. Practical Recommendations for Industry

Given the analysis presented so far, the results can be generalized within the following recommendations. The presented recommendations are essential to the developers or manufacturers of autonomous vehicles (inside the firewall) as well as to third-party contributors, such as Tier 1 companies (hopefully also from within the firewall). To the best of our knowledge, at this very moment, none of the OBE nor HSM specialists can declare that the cryptographic processor chosen for their system is able to manage the widespread highest standard for cryptographic algorithms (i.e. at least NIST level). Therefore, unless their OBE or HSM is explicitly mentioned in the following recommendations, there is a good chance that the related systems will not provide a high level of security on their own [10]. Guidance is more advanced in the case of recommendations aimed at inside-the-firewall autonomous vehicle development, in which three requirements are connected with data management protection (FDT, FMI/HSM, OTA-by-kills), while only one concerns only the manufacturing process.

Cybersecurity is extremely important in the development of new IoT systems and devices [7]. Cyber attacks can take many forms and can cause a broad range of consequences and a large financial impact. Cybersecurity is one of the most critical topics from a general IoT perspective, and takes on even greater importance considering the potential impact on autonomous vehicles. State-of-the-art solutions for Over-the-Air (OTA) software updates,

which are the focus of this work, typically focus on authenticity, integrity, and, in some cases, optional confidentiality [5]. The Remote Software Upload and Authorization Portal is a mechanism created in the automotive domain that can be used for secure software updates. Consequently, we can define the ROBUSTO requirements that are necessary to protect these two processes. This list includes the relevant cybersecurity recommendations, as well as detailed specification of cryptosystems that can be used to protect secret information. Finally, we specify the distinct features of the ROBUSTO ecosystem (related to software upload security and firmware management) that increase cybersecurity.

**References:**

1. [1] G. Abdelkader, K. Elgazzar, and A. Khamis, "Connected Vehicles: Technology Review, State of the Art, Challenges and Opportunities," 2021. ncbi.nlm.nih.gov
2. [2] Y. Shin and S. Jeon, "MQTree: Secure OTA Protocol Using MQTT and MerkleTree," 2024. ncbi.nlm.nih.gov
3. [3] S. Halder, A. Ghosal, and M. Conti, "Secure OTA Software Updates in Connected Vehicles: A survey," 2019. [PDF]
4. [4] C. Olarte, V. de Paiva, E. Pimentel, and G. Reis, "The ILLTP Library for Intuitionistic Linear Logic," 2019. [PDF]
5. [5] M. La Manna, L. Treccozzi, P. Perazzo, S. Saponara et al., "Performance Evaluation of Attribute-Based Encryption in Automotive Embedded Platform for Secure Software Over-The-Air Update," 2021. ncbi.nlm.nih.gov
6. Tatineni, Sumanth. "Cloud-Based Business Continuity and Disaster Recovery Strategies." *International Research Journal of Modernization in Engineering, Technology, and Science*5.11 (2023): 1389-1397.
7. Vemori, Vamsi. "Harnessing Natural Language Processing for Context-Aware, Emotionally Intelligent Human-Vehicle Interaction: Towards Personalized User Experiences in Autonomous Vehicles." *Journal of Artificial Intelligence Research and Applications* 3.2 (2023): 53-86.
8. Tatineni, Sumanth. "Security and Compliance in Parallel Computing Cloud Services." *International Journal of Science and Research (IJSR)* 12.10 (2023): 972-1977.

9. Gudala, Leeladhar, and Mahammad Shaik. "Leveraging Artificial Intelligence for Enhanced Verification: A Multi-Faceted Case Study Analysis of Best Practices and Challenges in Implementing AI-driven Zero Trust Security Models." *Journal of AI-Assisted Scientific Discovery* 3.2 (2023): 62-84.
10. [10] R. Singh Rathore, C. Hewage, O. Kaiwartya, and J. Lloret, "In-Vehicle Communication Cyber Security: Challenges and Solutions," 2022. ncbi.nlm.nih.gov
11. [11] F. Rosique, P. J. Navarro, C. Fernández, and A. Padilla, "A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research," 2019. ncbi.nlm.nih.gov
12. [12] P. Mundhenk, A. Paverd, A. Mrowca, S. Steinhorst et al., "Security in Automotive Networks: Lightweight Authentication and Authorization," 2017. [PDF]
13. [13] S. Saponara, S. Giordano, and R. Mariani, "Recent Trends on IoT Systems for Traffic Monitoring and for Autonomous and Connected Vehicles," 2021. ncbi.nlm.nih.gov
14. [14] M. Grosso, I. Cristinel Raileanu, J. Krause, M. Alonso Raposo et al., "How will vehicle automation and electrification affect the automotive maintenance, repair sector?," 2021. ncbi.nlm.nih.gov