Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

Future generation Security for smart and connected Cars - FuSeCar

Deliverable D3.1: Selection of candidate post-quantum algorithms

WP3: Post-quantum safety for current vehicular communication protocols and architectures

Authors:

Gianluca Dini[1]

gianluca.dini@unipi.it
[1]Dept. of Information Engineering
University of Pisa

Current revision: R1.1
Delivery date: 20th, October 2024

Revision history

Authors	Changes	Date	Revision
Gianluca Dini	Creation of the document, tentative structure	March 3rd, 2024	R0.1
Gianluca Dini	First draft of Section 1	March 20th, 2024	R0.2
Gianluca Dini	First draft of Section 2	April 14th, 2024	R0.3
Gianluca Dini	First draft of Section 4	May 17th, 2024	R0.4
Gianluca Dini	Draft of Section 5	July 18th, 2024	R0.6
Gianluca Dini	First draft of complete document	September 16th, 2024	R1.0
Gianluca Dini	Revision of document and minor fixes	October 20th, 2024	R1.1

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

Contents

1 Introduction

The automotive industry is undergoing a profound transformation as vehicles become increasingly digitalized to enable automation, connectivity, and shared mobility. Modern cars integrate up to 150 Electronic Control Units (ECUs) and approximately 100 million lines of code—four times more than a fighter jet—with projections of reaching 300 million lines by 2030.

Due to the growing complexity and connectivity of vehicles, Over-the-Air (OTA) software updates have become a cornerstone of modern automotive systems. Traditionally, software maintenance and upgrades required in-person servicing at dealerships, leading to high costs, long delays, and limited responsiveness to emerging issues. OTA updates eliminate these inefficiencies by enabling manufacturers to remotely deliver software patches, feature enhancements, and security fixes directly to vehicles. This capability is particularly important given the increasing reliance on software-defined functionality in areas such as infotainment, advanced driver-assistance systems (ADAS), and battery management in electric vehicles. Beyond improving user convenience, OTA updates strengthen cybersecurity resilience by allowing rapid deployment of security patches against evolving threats. As complex software inevitably contains vulnerabilities, and vehicles are widely connected via wireless networks, rapid distribution of software updates and security patches is essential. Finally, OTA updates support sustainability and long-term product value by extending the functional lifespan of vehicles without the need for physical recalls. Collectively, these factors make OTA updates not only a technical necessity but also a strategic enabler for innovation, safety, and customer satisfaction in the automotive industry.

Secure OTA software updates are critical in modern automotive systems because vehicles are increasingly connected, software-driven, and exposed to cyber threats. Unlike traditional updates performed in controlled service environments, OTA updates are delivered remotely over public networks, making them potential targets for interception, tampering, or unauthorized access. Secure OTA mechanisms ensure the authenticity, integrity, and confidentiality of software packages through cryptographic protections, preventing malicious code injection and guaranteeing that only authorized manufacturers can issue updates. A compromised update could enable attackers to disable safety features, disrupt vehicle operations, or steal sensitive data, posing risks not only to individual drivers but also to road safety and public trust.

To address these cybersecurity risks, the United Nations introduced two binding regulations: R155 on vehicle cybersecurity management and R156 on secure software updates [43, 44]. R155 references ISO/SAE 21434, which defines cybersecurity risk management requirements, while R156 specifies provisions for safe OTA software updates, ensuring integrity and authenticity of update mechanisms. Both regulations, mandatory in the EU since July 2022 for new vehicle types and from July 2024 for all new vehicles, represent a major challenge for manufacturers.

Secure OTA updates systems achieve authenticity and integrity through digital signatures. Unfortunately, conventional digital signature schemes based on RSA and ECC are vulnerable to quantum attacks, as Shor's algorithm could break them on sufficiently powerful quantum computers. In response, the National Institute of Standards and Technology (NIST) initiated a standardization process for quantum-resistant algorithms in 2016. By 2022, three digital signature schemes were selected, with draft standards published in 2023. NIST formally released its first set of post-quantum cryptography (PQC) standards on August 15, 2024. These standards include algorithms designed to resist attacks from both classical and quantum computers, addressing public-key encryption, key establishment, and digital signatures. Specifically, NIST standardized CRYSTALS-Kyber for public-key encryption and key encapsulation, as well as CRYSTALS-Dilithium, FALCON, and SPHINCS+ for digital signatures. This release marks the culmination of a multi-year international competition and evaluation process, establishing the first generation of quantum-resistant cryptographic standards to replace vulnerable classical algorithms such as RSA and ECC.

Secure OTA update systems typically prioritize authenticity and integrity while treating confidentiality as optional. However, without encryption, the intellectual property within updates—such as innovative security

measures or new features—remains unprotected, allowing competitors to intercept and analyse them during transmission. Even when secure channels (e.g., TLS) provide confidentiality while "in-transit," updates remain vulnerable when "at rest," such as when stored on untrusted third-party cloud servers without encryption. Manufacturers using their own trusted servers face another challenge: upon reaching a vehicle's internet-connected gateway ECU, an update can be intercepted if the gateway is tampered with, as it is more exposed to cyber-attacks than other ECUs. The AUTOSAR specification recommends a dedicated ECU, separate from the gateway, to manage updates securely [3]. A potential solution is to encrypt updates asymmetrically (e.g., using RSA), ensuring only this dedicated ECU can decrypt them. However, this approach is resource-intensive, requiring manufacturers to encrypt updates uniquely for each vehicle.

Prior works demonstrated that Ciphertext Policy Attribute-Based Encryption (CP-ABE) enhances in-transit and at-rest confidentiality of OTA software updates in automotive systems [15, 16, 26]. These works proved CP-ABE's seamless integration into existing OTA solutions, showing its computational and storage overhead is minimal compared to other OTA process costs, thus improving security at low expense. However, CP-AB schemes predominantly rely on pairing-based mathematics, which are vulnerable to large-scale quantum computer attacks, rendering them insecure in a future quantum landscape [38]. In the existing literature, there are several quantum-resistant CP-ABE schemes based on Ring Learning With Errors (RLWE) but most of them tend to require more computational resources than their quantum-weak counterparts [10, 17, 42, 45, 47, 46].

In this document we select post-quantum cryptographic algorithms to evaluate. These algorithms will be used to design and implement on automotive-oriented platforms, in a prototype form, a secure OTA software updates systems. The algorithms should be compliant with the NIST standardization context.[1]

The document is organized as follows. Section 2 briefly discusses related works. Section 3 introduces preliminary concepts, namely the NIST standard on post-quantum cryptography (Section 3.1) and attribute-based encryption (Section 3.2). Section 4 illustrates the reference model for OTA software updates, including the threat model, that we take as a reference. Section 5 states the post-quantum cryptographic algorithms that we choose for the rest of the project. Finally, Section **??** makes some concluding remarks.

[1] `https://csrc.nist.gov/projects/post-quantum-cryptography`.

2 Related Works

Recently, numerous schemes have been proposed to ensure secure OTA updates. To meet security objectives, researchers have employed secure software repository frameworks, blockchain, hash functions, and hybrid symmetric–asymmetric encryption.

Kuppusamy et al. [22, 25] introduced *Uptane*, a secure OTA framework for connected vehicles based on TUF [36]. Uptane introduces a two-tier architecture (director repository and image repossitory) and relies on multiple administrator roles, each signing update images and metadata with separate keys, to mitigate insider and key compromise attacks. Updates are verified by the primary ECU before installation in secondary ECUs, enhancing compromise resilience. However, it remains vulnerable to rollback attacks [2]. Uptane is the most widely referenced system in the automotive OTA literature and has been standardized by IEEE (IEEE 2943.1-2020) [20].

The ASSURED system focuses on resource-constrained embedded and automotive ECUs, extending existing work like Uptane [2]. ASSURED introduces i) end-to-end authenticity and integrity of update artifacts using digital signatures; ii) authorization delegation, where OEMs can delegate update approval to trusted third parties while retaining control; iii) lightweight cryptographic protocols to suit devices with limited resources; and, finally, iv) evidence-based updates, allowing devices to provide verifiable proof of correct installation to the backend. It is designed to meet regulatory demands (e.g., UN R155/R156) while keeping the system efficient and practical for in-vehicle ECUs.

A number of proposals exploit blockchain (BC) and smart contracts to address the security and privacy issues of OTA update for connected vehicles [41, 5, 48, 12]. Despite ensuring integrity and scalability, BC-based methods are resource-intensive, slow, and unable to guarantee source reliability.

Several hash function-based schemes have been proposed for securing OTA updates. In [32], updates are fragmented, linked through a reverse-order hash chain, and each fragment is encrypted with a pre-shared key before transmission. Similarly, [33] delivers encrypted updates with a hash-based verification code, while [31] employs a hash-based key derivation function. These methods are lightweight but face scalability challenges and are vulnerable to DoS attacks.

Combining symmetric and asymmetric techniques, Steger et al. [40] proposed *SecUp*, which uses RSA and session keys for secure communications. Other works integrate modified RSA with steganography to achieve multi-layered protection of updates stored in the cloud [29]. Although such methods enable parallel updates, they introduce high communication latency.

HSM-based protocols have been proposed to secure OTA updates by storing cryptographic keys and executing encryption operations [19]. Petri et al. [34] employed TPMs to validate updates against pre-stored hashes, while [13] used chip-specific features to fingerprint devices. These hardware-based schemes provide strong tamper resistance but lack support for parallel updates.

Differently from the previous works, La Manna et al. investigated the application of the Ciphertext Policy Attribute-Based Encryption (CP-ABE) scheme by Bethencourt et al. [7] to assure confidentiality of OTA software updates in automotive [27]. Attribute-Based Encryption (ABE) enhances efficiency in multiple-receiver end-to-end encryption and addresses the "update at rest" vulnerability by ensuring confidentiality. With ABE, updates remain encrypted and signed with long-term keys held by a dedicated ECU, rendering gateway tampering ineffective. An adversary can only access the update by compromising either the dedicated ECU or the target ECU requiring the update. Contribution of La Manna et al.'s work is threefold: (i) demonstrating an Attribute-Based Encryption (ABE) technique for secure Over-the-Air (OTA) software and firmware updates, designed for seamless integration with existing state-of-the-art solutions; (ii) establishing ABE's compatibility with the in-vehicle network architecture of contemporary vehicles and the computational constraints of authentic automotive Electronic Control Units (ECUs); and, finally, (iii) providing an experimental performance evaluation of ABE on a realistic automotive-compliant platform. Ghosal et al. proposed STRIDE, an OTA update

scheme for autonomous vehicles, employing the Bethencourt et al.'s CP-ABE to ensure confidentiality [15, 16]. Their performance evaluation, conducted via OMNeT++ simulations, is comprehensive but lacks testing on a real automotive platform.

3 Preliminaries

In Section 3.1 we briefly discuss the NIST standard whereas in Section 3.2 we introduce the notion of Attribute-based Encryption scheme.

3.1 The NIST Standard

In the last three decades, public key cryptography has become an indispensable component of our global communication digital infrastructure. Many of our most crucial communication protocols rely principally on three core cryptographic functionalities: public key encryption, digital signatures, and key exchange. Currently, these functionalities are primarily implemented using Diffie-Hellman key exchange, the RSA (Rivest-Shamir-Adleman) cryptosystem, and elliptic curve cryptosystems. The security of these depends on the difficulty of certain number theoretic problems such as Integer Factorization or the Discrete Log Problem over various groups [9].

In recent years, there has been a substantial amount of research on quantum computers—machines that exploit quantum mechanical phenomena to solve mathematical problems that are difficult or intractable for conventional computers. If large-scale quantum computers are ever built, they will be able to break many of the public-key cryptosystems currently in use. In 1994, Peter Shor of Bell Laboratories showed that quantum computers, a new technology leveraging the physical properties of matter and energy to perform calculations, can efficiently solve each of these problems, thereby rendering all public key cryptosystems based on such assumptions impotent [37, 39]. Thus, a sufficiently powerful quantum computer will put many forms of modern communication—from key exchange to encryption to digital authentication—in danger [9]. This would seriously compromise the confidentiality and integrity of digital communications on the Internet and elsewhere. The goal of post-quantum cryptography (also called quantum-resistant cryptography) is to develop cryptographic systems that are secure against both quantum and classical computers and can interoperate with existing communications protocols and networks.

Algorithm	**Type**	**Purpose**	**Impact**
AES	Symmetric Key	Encryption	Larger key sizes are needed
SHA-2, SHA-3	-	Hash function	Larger output needed
RSA	Public key	Digital signature, key establishment	No longer secure
ECDSA, ECDH	Public key	Digital signature, key establishment	No longer secure
DSA	Public key	Digital signatures, key establishment	No longer secure

Table 1: Impact of quantum computing on common cryptographic algorithms.

In the twenty years since Shor's discovery, the theory of quantum algorithms has developed significantly. Quantum algorithms achieving exponential speedup have been discovered for several problems relating to physics simulation, number theory, and topology. Nevertheless, the list of problems admitting exponential speedup by quantum computation remains relatively small. In contrast, more modest speedups have been developed for broad classes of problems related to searching, collision finding, and evaluation of Boolean formulae. Grover's search algorithm proffers a quadratic speedup on unstructured search problems. While such a speedup does not render cryptographic technologies obsolete, it can have the effect of requiring larger key sizes, even in the symmetric key case. See Table 1 for a summary of the impact of large-scale quantum computers on common cryptographic algorithms, such as RSA and the Advanced Encryption Standard (AES). It is not known how far these quantum advantages can be pushed, nor how wide is the gap between feasibility in the classical and

quantum models [9].

The question of when a large-scale quantum computer will be built is complicated and contentious. While in the past it was less clear that large quantum computers are physical possible, many scientists now believe it to be merely a significant engineering challenge. Some experts even predict that within the next 20 or so years, sufficiently large quantum computers will be built to break essentially all public key schemes currently in use. It has taken almost 20 years to deploy our modern public key cryptography infrastructure. It will take significant effort to ensure a smooth and secure migration from the current widely used cryptosystems to their quantum computing resistant counterparts. Therefore, regardless of whether we can estimate the exact time of the arrival of the quantum computing era, we must begin now to prepare our information security systems to be able to resist quantum computing [9].

A large international community has emerged to address the issue of information security in a quantum computing future, in the hope that our public key infrastructure may remain intact by utilizing new quantum-resistant primitives. Initiatives have been started at academic and industry level and have received support from national funding agencies. Notably, the European Union (EU) has funded projects PQCrypto[2] and SAFEcrypto.[3] In the past few years standard organizations have started their own activities. More specifically, since 2013, the European Telecommunications Standards Institute (ETSI) has held three "Quantum-Safe Cryptography" workshops, and in 2015 NIST held a workshop on "Cybersecurity in a Post-Quantum World," which was attended by over 140 people from government, industry, and academia. NIST issued a public call for submissions to the Post-Quantum Cryptography (PQC) Standardization Process in December 2016. Prior to the November 2017 deadline, a total of 82 candidate algorithms were submitted. After a year-long review of the candidates, NIST selected 26 algorithms to move on to the second round of evaluation in January 2019. Then, NIST selected seven finalists and eight alternates to move on to the third round of evaluation in July 2020. The third round of evaluation began in July 2020 and continued for approximately 18 months. After three rounds of evaluation and analysis, NIST selected four algorithms it will standardize because of the PQC Standardization Process:

- CRYSTALS-Kyber, a key encapsulation mechanism (KEM) selected for general encryption, such as for accessing secured websites.
- CRYSTALS-Dilithium and FALCON, two lattice-based algorithms chosen for general-purpose digital signature protocols.
- SPHINCS+, a stateless hash-based digital signature scheme.

Finally, on September 2024, NIST released the final set of encryption tools designed to withstand the attack of a quantum computer. The National Security Memorandum 10 (NSM-10) establishes the year 2035 as the primary target for completing the migration to PQC across Federal systems [NSM10]. In the Internal Report 8547, NIST delineates a more precise transition roadmap [NIS24]. For example, as to Digital Signatures (ECDSA and RSA) and Key Establishment Schemes (DH, ECDH, and RSA), the 112-bit security level will be deprecated after 2030 and disallowed after 2035, whereas the 128-bit security level will be disallowed after 2035.

3.2 Attribute-based Encryption

Attribute-Based Encryption (ABE) is a cryptographic paradigm that extends traditional public-key encryption by enabling fine-grained access control over encrypted data. Unlike conventional encryption schemes where a specific recipient's public key is used, ABE associates ciphertexts and keys with *attributes* or *policies*, allowing decryption only when predefined conditions are met. Two prominent variants of ABE are *Ciphertext-Policy*

[2] https://pqcrypto.eu.org/
[3] https://www.safecrypto.eu/

Attribute-Based Encryption (CP-ABE) and *Key- Policy Attribute-Based Encryption* (KP-ABE), each distinguished by how access control is enforced.

In CP-ABE, the access policy is embedded within the ciphertext by the encryptor during the encryption process. Each private key is associated with a set of attributes that describe the key holder's properties or roles. Decryption is possible only if the attributes linked to the private key satisfy the access policy specified in the ciphertext. In KP-ABE, the access policy is embedded within the private key, while the ciphertext is associated with a set of attributes. The encryptor labels the ciphertext with descriptive attributes, and a user can decrypt it only if the policy in their private key is satisfied by the attributes of the ciphertext. In this work, we refer to CP-ABE.

A CP-ABE system comprises four fundamental algorithms, setup, encryption, key generation, and decryption. Their specification follows.

The *Setup* algorithm $(EK, MK) \leftarrow Setup(1^n)$ accepts a security parameter n as input, which denotes the desired level of cryptographic robustness against potential attacks. This parameter governs the computational complexity of the cryptographic operations and the size of the generated keys and ciphertexts. The algorithm outputs the *public parameter* EK and a *master key* MK.

The *Encryption* algorithm $y \leftarrow Encrypt(EK, x, A)$ takes as input the public parameters EK, a cleartext x, and an access structure A defined over the universe of attributes. The algorithm encrypts the cleartext x and produces a ciphertext y such that only a user possessing a set of attributes that satisfies the access structure will be able to decrypt the message. It is assumed that the ciphertext implicitly contains A.

The *Key Generation* algorithm $DK \leftarrow KeyGen(MK, S)$ takes as input the master key MK and a set of attributes S that describe the key and returns a private decryption key DK.

The *Decryption* algorithm $x \leftarrow Decrypt(PK, y, SK)$ takes as input the public parameters EK, a ciphertext y containing an access policy A, and a private decryption key DK described by a set of attributes S. If the set of attributes S satisfies the access structure A, the algorithm decrypts the ciphertext and returns the cleartext x; otherwise, the algorithm fails and returns *null*.

By default, CP-ABE encryption is performed using the *digital envelope* technique. This means that the CP-ABE ciphertext protects a symmetric key, which is used to symmetrically encrypt the actual ciphertext. We will consider the Advanced Encryption Standard (AES) to be the symmetric cipher. Unless otherwise specified, from now on, we will assume that CP-ABE encryption is performed using the digital envelope technique.

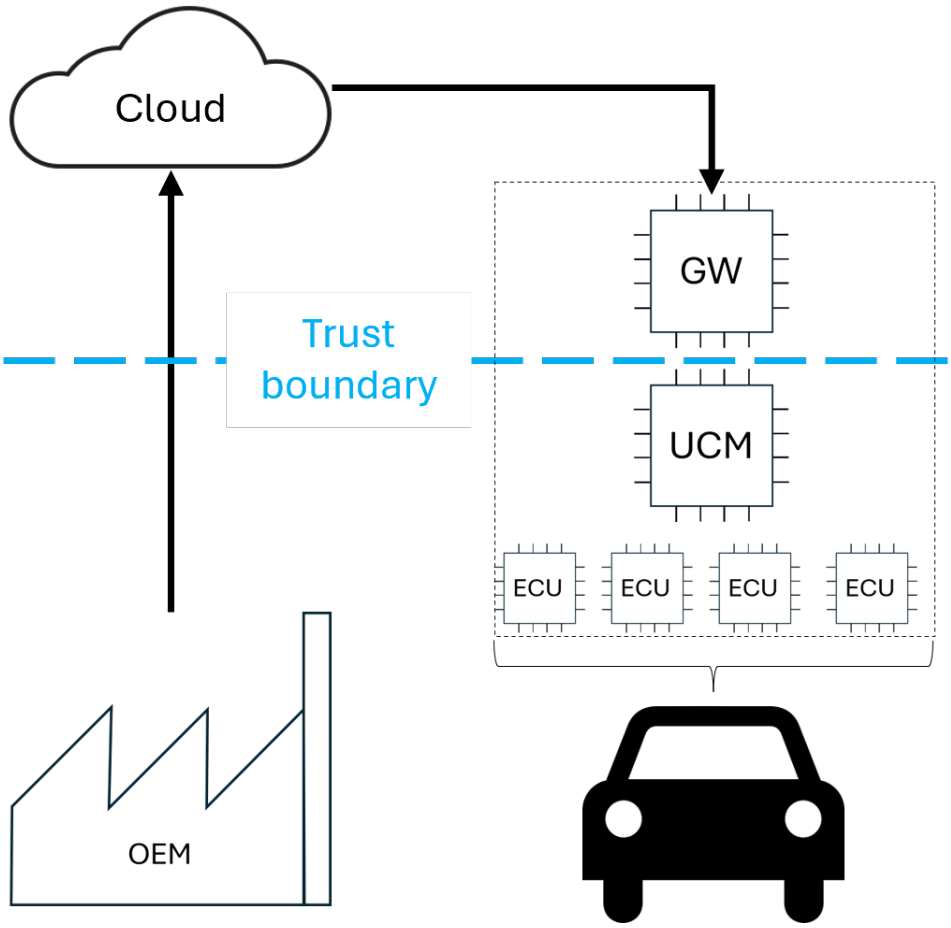

Figure 1: Abstract reference Model of an OTA update system.

4 System Model

Designing an automotive FOTA system is not an easy task, because it must take into consideration numerous threat agents and deal with many possible attacks. In order to give to our reference FOTA system more realism, we designed it to be compliant to Uptane2 by Kuppusamy et al. [25, 23], which is an industry recognized IEEE-ISTO standard [20] and it is now incorporated in the Automotive Grade Linux operating system[4].

With reference to Figure 1, we consider an OTA update system which features many vehicles, an Origina Equipment Manufacturer (OEM), and an *honest-but-curious* cloud server (CS). The manufacturer is in charge of generating all of the cryptographic keys needed in the system. The manufacturer possesses i) a signing key SK to sign updates, ii) the CP-ABE master key MK, and iii) the CP-ABE encryption key EK for update encryption.

We assume that each vehicle has an ECU dedicated to OTA updates called *Update and Configuration Manager* (UCM), as specified in the AUTOSAR specification document [3]. This ECU is not connected directly to the internet, rather it is connected to both an ECU which plays the role of a *network gateway* (GW) and each ECU that supports the OTA update functionality.

Each vehicle possesses the following cryptographic keys. A CP-ABE decryption key DK to decrypt encrypted updates. The decryption keys embeds a set of attributes that describes the vehicle's components and characteristics. A vehicle also possesses the manufacturer's public verification key VK to verify digital signa-

[4] `https://www.automotivelinux.org/`

tures on updates. These vehicle-related cryptographic keys are installed in the UCM by the OEM at the time of its construction.

The manufacturer OEM is in charge to produce the software update, encrypt it with the CP-ABE scheme, sign it along with a version number, and store the signed and encrypted update on the cloud server CS. This server sends the signed and encrypted update to any vehicle that requests it.

Upon reception of the software update, the vehicle gateway GW forwards the message to the UCM, which verifies the manufacturer signature, and decrypts the CP-ABE ciphertext. Finally, the UCM forwards the software update to the intended ECU, which installs it as soon as the user gives his/her consent.

4.1 The Threat Model

We consider both external and internal adversaries.

For protocol analysis, in general, the most well-known adversary model is the so-called Dolev-Yao (DY) model [11]. The DY model is also one of the strongest possible adversaries in terms of capabilities. In the ideal case, security and privacy properties would be maintained even against a DY adversary. Intuitively, consider a network where messages are constantly being sent between different parties. In the DY model, we assume there exists an adversary who has significant control over the communication channel and, consequently, can eavesdrop messages as well as intercept, modify, force, replay and delete them. The only limitation on the adversary is the strength of the cryptographic methods used. For example, if a message is encrypted, the adversary cannot decrypt it unless they can break the encryption algorithm. As a further example, if the message is digitally signed, the adversary cannot modify it, unless they can break the digital signature algorithm. This model is powerful because it assumes the worst-case scenario, making it a robust framework for testing the security of protocols. By designing protocols that can withstand such an adversary, we can ensure a high level of security in real-world applications.

While the DY model is adequate for modeling external adversaries, however, in certain cases, a DY adversary is too strong to be used in a realistic model of the system. In an automotive system, for instance, the system should be secure against an external DY adversary. However, a legitimate member of the system model could not realistically be modeled as a DY adversary. In reality, various factors limit the capabilities of internal members including regulations, audits, oversight and desire to maintain reputation. However, although a DY model is not appropriate in this case, it does not necessarily mean that one of the above actors is not adversarial. We therefore assume that these agents are *semi-honest* or *honest-but-curious* which we define as follows: the honest but-curious (HBC) adversary is a legitimate participant to a system who will not deviate from the specification but will attempt to learn all possible information from legitimately received messages or stored data. In comparison to the DY model, the HBC model is more limited in that an adversary will not deviate from the protocol and cannot send any falsified messages. Even in comparison to a passive DY adversary, the HBC adversary is still more limited in that it cannot eavesdrop on arbitrary communication channels and can only receive messages of which it is the intended recipient

Coming back to the Reference Model in Figure 1, we assume that the manufacturer OEM and update and the configuration manager UCM are trusted, whereas the in-vehicle gateway GW and the Cloud Server CS are honest-but-curious. This implies that we assumne an adversary that is capable of breaching these components and stealing all the data stored in them but does not alter their behaviour in correctly executing all the protocols, basically because the adversary tries to remain as stealth as possible during the attack. Note that this reflects real-life attacks against cloud servers that leverage some common weakness, for example buffer overflows or code injections, or hardware vulnerabilities like Meltdown or Spectre [28, 24].

This is a reasonable assumption also for the the in-vehicle gateway as it is the only ECU directly connected to the internet and is thus more vulnerable to external attacks than other ECUs [26, 18, 8, 21].

5 Selection of Post-Quantum Cryptographic Algorithms

In this project, our task consists in evaluating the impact of Post-Quantum Cryptography on secure OTA update system. In order to do that we consider the set of Post-Quantum cryptographic algorithms mentioned below. In any case, in our evaluation, we focus on the vehicle side, which has limited computing resources. For our benchmarks, as an experimental automotive board, we use an FPGA Xilinx ZCU106 evaluation board, which is usually used for prototypes in the automotive field.

As to authenticity, we consider two of the three standardized Post-Quantum Cryptography (PQC) digital signature algorithms, namely Crystals-Dilithium [4] (hereafter "Dilithium" for brevity) and Falcon [14], which are both based on the efficient structured-lattice mathematics. We leave out the third one, Sphincs+ [30, 6], because it has considerably worse performance than Dilithium and Falcon. Indeed, Sphincs+ is not based on lattices, and NIST selected it only as a fallback option, in case of a major breakthrough in lattice cryptanalysis should occur [1].

As to confidentiality, we consider the CP-ABE scheme based on Ring Learning with Errors (RLWE) proposed by Gür et al. [17], which adapts the ring-based approach of Zhang et al. [49]. This scheme is believed to be resistant to quantum attacks, so it is a candidate replacement of pairing-based ABE schemes in the future quantum world. To maintain conciseness, we omit a comprehensive description of the scheme. Readers seeking a complete exposition are directed to [17].

Gür et al. implemented the CP-ABE scheme in Palisade[5] and made it available for community use. Palisade is an open-source C++ library that provides implementations of various lattice-based cryptographic primitives and schemes.

Palisade allows us to express access policies using a method based on *string pattern* [35]. The access policies are composed by a series of AND logical operations between attributes, each of which may be preceded by a NOT operator. The set of the attributes that appear without the NOT inside the policy are said to be *affirmed attributes*. Conversely, those that appear with the NOT are said *negated attributes*. An example of access policy is the following one: attr1 AND attr2 AND NOT attr3, where attr1 and attr2 are affirmed attributes while attr3 is negated. The attributes of the universe that do not appear in the policy are said *don't-care attributes*, since their logical value does not influence the outcome of the policy.

While primarily designed for homomorphic encryption, Palisade also supports post-quantum public key encryption, digital signatures, proxy re-encryption, multiparty computation, identity-based encryption, and attribute-based encryption (ABE). As of this writing, Palisade is, to our knowledge, the sole library offering Ciphertext-Policy Attribute-Based Encryption (CP-ABE) with post-quantum security.

[5] `https://palisade-crypto.org`

References

[1] Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Carl Miller, Dustin Moody, Rene Peralta, et al. Status report on the third round of the NIST post-quantum cryptography standardization process. *US Department of Commerce, NIST*, 2022.

[2] N Asokan, Thomas Nyman, Norrathep Rattanavipanon, Ahmad-Reza Sadeghi, and Gene Tsudik. ASSURED: Architecture for secure software update of realistic embedded devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 37(11):2290–2300, 2018.

[3] AUTOSAR. Specification of update and configuration management. `https://www.autosar.org/fileadmin/standards/R21-11/AP/AUTOSAR_SWS_UpdateAndConfigurationManagement.pdf`, 2021. Accessed: 2025-05-20.

[4] Shi Bai, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS-dilithium: Algorithm specifications and supporting documentation. Technical report, 2020.

[5] Mohamed Baza, Mahmoud Nabil, Noureddine Lasla, Kemal Fidan, Mohamed Mahmoud, and Mohamed Abdallah. Blockchain-based firmware update scheme tailored for autonomous vehicles. In *2019 IEEE wireless communications and networking conference (WCNC)*, pages 1–7. IEEE, 2019.

[6] Daniel J. Bernstein, Andreas Hülsing, Stefan Kölbl, Ruben Niederhagen, Joost Rijneveld, and Peter Schwabe. The sphincs+ signature framework. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, pages 2129–2146, New York, NY, USA, 2019. Association for Computing Machinery.

[7] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *2007 IEEE Symposium on Security and Privacy (SP '07)*, pages 321–334, 2007.

[8] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. Comprehensive experimental analyses of automotive attack surfaces. In *20th USENIX security symposium (USENIX Security 11)*, 2011.

[9] Lily Chen, Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray A Perlner, and Daniel Smith-Tone. Report on Post-Quantum Cryptography. Technical Report NISTIR 8105, US Department of Commerce, National Institute of Standards and Technology, April 2016.

[10] Marco Cianfriglia, Elia Onofri, and Marco Pedicini. mrlwe-cp-abe: A revocable cp-abe for post-quantum cryptography. *Journal of Mathematical Cryptology*, 18(1):20230026, 2024.

[11] Danny Dolev and Andrew Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 2003.

[12] Ali Dorri, Marco Steger, Salil S Kanhere, and Raja Jurdak. Blockchain: A distributed solution to automotive security and privacy. *IEEE communications magazine*, 55(12):119–125, 2017.

[13] Solon Falas, Charalambos Konstantinou, and Maria K Michael. A modular end-to-end framework for secure firmware updates on embedded systems. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 18(1):1–19, 2021.

[14] Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. Falcon: Fast-fourier lattice-based compact signatures over NTRU specification v1.2. Technical report, 2020.

[15] Amrita Ghosal, Subir Halder, and Mauro Conti. Stride: Scalable and secure over-the-air software update scheme for autonomous vehicles. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pages 1–6, 2020.

[16] Amrita Ghosal, Subir Halder, and Mauro Conti. Secure over-the-air software update for connected vehicles. *Computer Networks*, 218:109394, 2022.

[17] Kamil D. Gür, Yuriy Polyakov, Kurt Rohloff, Gerard W. Ryan, Hadi Sajjadpour, and Erkay Savaş. Practical applications of improved gaussian sampling for trapdoor lattices. *IEEE Transactions on Computers*, 68(4):570–584, 2019.

[18] Numaan Huq, Craig Gibson, and Rainer Vosseler. Driving security into connected cars: threat model and recommendations. Technical report, Trend Micro Research, 2020.

[19] Muhammad Sabir Idrees, Hendrik Schweppe, Yves Roudier, Marko Wolf, Dirk Scheuermann, and Olaf Henniger. Secure automotive on-board protocols: A case of over-the-air firmware updates. In *Communication Technologies for Vehicles: Third International Workshop, Nets4Cars/Nets4Trains 2011, Oberpfaffenhofen, Germany, March 23-24, 2011. Proceedings 3*, pages 224–238. Springer, 2011.

[20] IEEE-ISTO. Uptane standard for design and implementation 2.1.0. https://uptane.org/docs/2.1.0/standard/uptane-standard, 2023. [Online; accessed April 20, 2024].

[21] Pengfei Jing, Zhiqiang Cai, Yingjie Cao, Le Yu, Yuefeng Du, Wenkai Zhang, Chenxiong Qian, Xiapu Luo, Sen Nie, and Shi Wu. Revisiting automotive attack surfaces: A practitioners' perspective. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2348–2365. IEEE, 2024.

[22] Trishank Karthik, Akan Brown, Sebastien Awwad, Damon McCoy, Russ Bielawski, Cameron Mott, Sam Lauzon, André Weimerskirch, and Justin Cappos. Uptane: Securing software updates for automobiles. In *International conference on embedded security in car*, volume 11, 2016.

[23] Trishank Karthik, Akan Brown, Sebastien Awwad, Damon McCoy, Russ Bielawski, Cameron Mott, Sam Lauzon, André Weimerskirch, and Justin Cappos. Uptane: Securing software updates for automobiles. In *International conference on embedded security in car*, volume 11, 2016.

[24] Paul Kocher, Jann Horn, Anders Fogh, Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, et al. Spectre attacks: Exploiting speculative execution. *Communications of the ACM*, 63(7):93–101, 2020.

[25] Trishank Karthik Kuppusamy, Lois Anne DeLong, and Justin Cappos. Uptane: Security and customizability of software updates for vehicles. *IEEE vehicular technology magazine*, 13(1):66–73, 2018.

[26] Michele La Manna, Luigi Treccozzi, Pericle Perazzo, Sergio Saponara, and Gianluca Dini. Performance evaluation of attribute-based encryption in automotive embedded platform for secure software over-the-air update. *Sensors*, 21(2), 2021.

[27] Michele La Manna, Luigi Treccozzi, Pericle Perazzo, Sergio Saponara, and Gianluca Dini. Performance evaluation of attribute-based encryption in automotive embedded platform for secure software over-the-air update. *Sensors*, 21(2), 2021.

[28] Moritz Lipp, Michael Schwarz, Daniel Gruss, Thomas Prescher, Werner Haas, Jann Horn, Stefan Mangard, Paul Kocher, Daniel Genkin, Yuval Yarom, et al. Meltdown: Reading kernel memory from user space. *Communications of the ACM*, 63(6):46–56, 2020.

[29] Kathiresh Mayilsamy, Neelaveni Ramachandran, and Vismitha Sunder Raj. An integrated approach for data security in vehicle diagnostics over internet protocol and software update over the air. *Computers & Electrical Engineering*, 71:578–593, 2018.

[30] National Institute of Standards and Technology. Stateless hash-based digital signature standard. Technical Report Federal Information Processing Standards Publications (FIPS) NIST FIPS 205 ipd, U.S. Department of Commerce, Washington, D.C., 2023.

[31] Vladimir Nikic, Dusan Bortnik, Milan Lukic, and Ivan Mezei. Firmware updates over the air using nb-iot wireless technology. In *2021 29th Telecommunications Forum (TELFOR)*, pages 1–4. IEEE, 2021.

[32] Dennis K Nilsson and Ulf E Larson. Secure firmware updates over the air in intelligent vehicles. In *ICC Workshops-2008 IEEE International Conference on Communications Workshops*, pages 380–384. IEEE, 2008.

[33] Dennis K Nilsson, Lei Sun, and Tatsuo Nakajima. A framework for self-verification of firmware updates over the air in vehicle ecus. In *2008 IEEE Globecom Workshops*, pages 1–5. IEEE, 2008.

[34] Richard Petri, Markus Springer, Daniel Zelle, Ira McDonald, Andreas Fuchs, and Christoph Krauß. Evaluation of lightweight tpms for automotive software updates over the air. In *Proc. of 4th International Conference on Embedded Security in Car USA*, pages 1–15, 2016.

[35] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), September 2009.

[36] Justin Samuel, Nick Mathewson, Justin Cappos, and Roger Dingledine. Survivable key compromise in software update systems. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 61–72, 2010.

[37] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[38] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[39] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[40] M Steger, CA Boano, T Niedermayr, M Karner, J Hillebrand, K Roemer, and W Rom. An integrated approach for data security in vehicle diagnostics over internet protocol and software update over the air. *IEEE Trans. on Industrial Informatics*, 14(5):2181–2193, 2018.

[41] Marco Steger, Ali Dorri, Salil S Kanhere, Kay Römer, Raja Jurdak, and Michael Karner. Secure wireless automotive software updates using blockchains: A proof of concept. In *Advanced Microsystems for Automotive Applications 2017: Smart Systems Transforming the Automobile*, pages 137–149. Springer, 2017.

[42] Jianguo Sun, Yuqing Qiao, Zechao Liu, Yitao Chen, and Yang Yang. Practical multi-authority ciphertext policy attribute-based encryption from r-lwe. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 1435–1443. IEEE, 2021.

[43] UNECE. UN Regulation No. 155 - Cyber security and cyber security management system. `https://unece.org/transport/documents/2021/03/standards/un-regulation-no-155-cyber-security-and-cyber-security`, 2021.

[44] UNECE. UN Regulation No. 156 - Software update and software update management system. `https://unece.org/transport/documents/2021/03/standards/un-regulation-no-156-software-update-and-software-update`, 2021.

[45] Yang Yang, Jianguo Sun, Zechao Liu, and YuQing Qiao. Practical revocable and multi-authority cp-abe scheme from rlwe for cloud computing. *Journal of Information Security and Applications*, 65:103108, 2022.

[46] Yun-Fei Yao, Hui-Yan Chen, You Gao, Ke Wang, and Hao-Yang Yu. A decentralized multi-authority cp-abe scheme from lwe. *Journal of Information Security and Applications*, 82:103752, 2024.

[47] Yunfei Yao, Huiyan Chen, Linzhi Shen, Ke Wang, and Qingnan Wang. A cp-abe scheme based on lattice lwe and its security analysis. *Applied Sciences*, 13(14):8043, 2023.

[48] Sadia Yeasmin and Anwar Haque. A multi-factor authenticated blockchain-based ota update framework for connected autonomous vehicles. In *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, pages 1–6. IEEE, 2021.

[49] Jiang Zhang, Zhenfeng Zhang, and Aijun Ge. Ciphertext policy attribute-based encryption from lattices. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '12, page 16–17, New York, NY, USA, 2012. Association for Computing Machinery.