Feasibility and Benchmarking of Post-Quantum Cryptography in the Cooperative ITS Ecosystem

Brigitte Lonc, Alexandre Aubry, Hafeda Bakhti, Maria Christofi, Hassane Aissaoui Mehrez

Feasibility and Benchmarking of Post-Quantum Cryptography in the Cooperative ITS Ecosystem

Brigitte Lonc
B. Lonc consulting / IRT SystemX
Palaiseau, France
brigitte.lonc@ext.irt-systemx.fr

Alexandre Aubry
Stellantis / IRT SystemX
Poissy, France
alexandre.aubry@stellantis.com

Hafeda Bakhti
Digital ID Atos / IRT SystemX
Puteaux, France
hafeda.bakhti@atos.net

Maria Christofi
Oppida / IRT SystemX
Montigny-le-Bretonneux, France
maria.christofi@oppida.fr

Hassane Aissaoui Mehrez
LTCI IMT - Telecom Paris
IP Paris / IRT SystemX
Palaiseau, France
hassane.aissaoui@telecom-paris.fr

***Abstract*—Localized communication between vehicles and their surrounding environment (V2X) is a key technology to enable Cooperative Intelligent Transportation Systems (C-ITS) aiming at road safety, traffic flow and driving comfort. Security services based on Elliptic Curve Cryptography (ECC) for authenticity and confidentiality (mostly application-dependent) have been chosen to meet the hard constraints of low latency safety communications and limited bandwidth radio communication in dense traffic conditions. Due to threats raised by Quantum Computers (QC), the classical asymmetric cryptographic algorithms could be broken impacting the Public Key Infrastructure (PKI)-based security solutions, with negative safety consequences on the (semi)-autonomous vehicles and road users. Our project (TAM: Trusted Autonomous Mobility) [18] is focusing on end-to-end cybersecurity and privacy for innovative services in the field of cooperative, connected and automated mobility (CCAM). One main objective is to find suitable quantum safe schemes to replace the current cryptographic standards based on ECC which are used in V2X communications. After defining the main requirements and key performance indicators for C-ITS, a benchmarking of current NIST pre-standards PQC algorithms was performed to assess the feasibility and performances in C-ITS applications and based on the results a best fit solution is selected.**

Keywords—C-ITS, security, KEM, post-quantum cryptography, performance analysis

I. Introduction

Quantum computers already exist today: they are still small-scale, but it is expected that large-scale QC will become a reality in the near future [1]. This emerging technology is a two-edged sword: one that increases the potential for technical evolution and the other that introduces new possibilities to attack modern cryptography by solving hard mathematical problems considered intractable by classical computers [2].

Even if symmetric cryptography is considered as resistant in a post-quantum era requiring "only" to adapt the key length (doubling seems sufficient), this is not the case for the asymmetric cryptography, as Shor's algorithm (published in 1997) allows to solve problems such as the integer factorization and the discrete logarithm in polynomial time. As for today, these problems are considered as hard and the security of many mechanisms is based on them; e.g. public-key encryption, key exchange schemes such as Diffie-Hellman and digital signatures (e.g. ECDSA). It is estimated that 4098 qubits will be necessary to tackle RSA2048 while only 2042 qubits for ECDSA224 meaning that ECC may be an easier target ([3], [4]). Asymmetric algorithms need thus to be replaced and to do so, NIST launched a Post-Quantum Cryptography (PQC) competition. In July 2022, the first milestone has been reached with a selection of three signature algorithms and one Key Encapsulation Mechanism (KEM) [5]. The corresponding standards are expected to be available in 2024. Other KEM candidates are still in the competition and a second call is launched to provide signatures schemes with "short signature and fast verification".

In a C-ITS context, cooperation between participants can only be established using a trusted, secure connectivity and this is currently achieved using messages' signature and a certificate management infrastructure (PKI).

Unfortunately, all PQC candidates have larger key, signature or cipher text size, which impact the size of the transmitted data on the network and the increased delay due to cryptographic operations is very challenging considering the strong latency requirements of safety-critical applications. Using these candidates PQC schemes, the security overhead would increase substantially and may exceed the maximum packet size ("MTU"). Then packet segmentation might be necessary, introducing additional steps in the security protocols which may not be compatible with the low latency requirements.

In this study, we intend to precisely identify the priority requirements that shall be fulfilled when replacing classic cryptographic by post-quantum algorithms and to evaluate the feasibility of such migration as well as the performance impacts on current security protocol standards supporting C-ITS legacy and emerging applications. This should help to decide the replacement strategy and target an acceptable trade-off between security, performance and network efficiency in order to improve future V2X communication technologies.

II. Related work

The performance issues and the constraints that the new PQC algorithms will add to existing solutions are major concerns. NIST has published a report with the performance benchmark of algorithms under competition [5]. Two platforms were selected as reference implementations: the Intel x86 (with AVX2 instruction set extension) that is extremely common and accessible on modern computers, laptops and servers, and the

ARM Cortex M4 (192kbytes of RAM, 1xxMHZ) which is more common in small IoT devices and have restricted performances and memory. Many publications also present benchmark results on other platforms such as ARM Cortex A, RISC-V or even hardware-based implementation on FPGAs. Research papers are providing efficient implementations and improved performance results on the target devices. For instance, [6] has investigated performances of NIST candidates on an automotive micro-controller.

In [7], N. Bindel et al. presented a security and performances comparison of post-quantum algorithms for vehicle-to-vehicle (V2V) communications. They focus on the BSM (Basic Safety Messages) standard used in US which defines the data and message sets similar to CAM/DENM standards for short-range communications of CAVs supporting vehicle safety applications. However, the paper is only focusing on message signature algorithms. The authors introduce the "pure PQ V2V design" that consists of sending PQ certificate in several fragments before being able to verify the received BSMs: this solution is less secure than the ECDSA-based design (as it leads to a period where the received BSMs cannot be authenticated) but allows to cope with large security overhead (public key and certificate). They propose three alternative designs for hybrid ECDSA and PQ schemes: the "True Hybrid" using the concatenation of ECDSA and PQ signatures and certificates, the "Backwards-compatible Hybrid" using also two signatures but the receiver can verify or not the PQ signature and the "Partially PQ Hybrid" where BSMs are signed using ECDSA and the integrity of the ECDSA signature keys is guaranteed by both ECDSA and PQ signatures.

It is worth noting that for starting the migration to PQC, several governmental agencies such as ENISA [2], NIST [8], ANSSI [9] and BSI [10] recommend using hybrid modes for signatures and KEMs. For signatures, this consists of using at least two cryptographic schemes simultaneously; which has many advantages but is affecting the performance of the protocols and applications as this may increase the required radio bandwidth and the needed resources on the target platforms (memory and CPU).

III. Feasibility assessment of PQC schemes for C-ITS

A. Background on V2X localized communications for C-ITS

C-ITS aim at providing safer, cleaner and more efficient transport services. These systems are designed around a common data communication and management architecture (the ITS station architecture) to share data between vehicles and the road infrastructure/road users, and to offer mobility services in the edge or cloud network. The architecture defines rules for this sharing of data and combines multiple radio access technologies, protocols, and functionalities to manage security, data and communications. Basically, the ITS station architecture comprises short-range localized communication technologies that allow the direct exchange of data between vehicles and their surrounding environment (V2X). By relying on secure, trusted communications and cooperation between the vehicles, the road infrastructure and the road users, C-ITS services are enabling the deployment of critical road safety applications and the building of future cooperative, connected and automated mobility (CCAM).

In a first step, Day1 use cases focusing on driver awareness and information were defined, using standardized messages such as Cooperative Awareness Messages (CAM) [11]. CAM which contain vehicle kinematic data are periodically transmitted by vehicles at variable frequency (1 to 10 Hertz). Based on ETSI and CEN/ISO standards, the system profile specifications were developed for ITS stations in vehicles or road-side equipment by car manufacturers/road operators' stakeholders in Car2Car Communication Consortium & C-ROADS ([12], [13]). Day1 use cases were field-tested and are now under deployment in Europe.

With the development of always more efficient communications technologies to support C-ITS services, advanced use cases are emerging, such as cooperative driving services or collective perception (allowing the sharing of perception sensor data between the vehicle or infrastructure's ITS stations). All these new services will benefit to the connected, autonomous vehicles (CAV). To support these advanced use cases, new connectivity technologies are being developed to support C-ITS deployment. In Europe, ETSI's ITS-G5 radio technology is considered as a promising solution for short-range communications. It is the European variant of IEEE 802.11p [28] (known as DSRC/WAVE), and an enhanced version, IEEE 802.11bd, is under development. Also, 3GPP has developed the LTE-V2X or C-V2X (cellular-V2X) technology which combines long- and short-range communications in LTE (in releases 14 and 15) and is further developing new radio communication specifications in 5G (Release 16) named 5G-V2X or NR-V2X to support advanced driving applications.

B. C-ITS trust model and security services in Europe

To secure C-ITS services based on V2X localized short-range communications between vehicles and their surrounding environment, standardization bodies such as ETSI and IEEE use asymmetric cryptography. This requires setting up a PKI to produce digital certificates of the ITS Stations Unit (ITS-S) deployed in vehicles On-Board Units (OBUs) and road-side units (RSUs) [14].

The PKI delivers two types of certificates to the ITS-S. The first type is called the Enrolment Certificate (EC) or long-term certificate. The EC is not directly used to secure C-ITS services based on V2X messages but is used as a proof of authentication of the ITS-S to request multiple pseudonym certificates named Authorization Tickets (ATs) or short-term certificate. ATs are used by the vehicle to sign its messages so the receiver can validate the message integrity, the sender's authenticity and authorization (application permissions). In order to protect driver's privacy, the vehicles change frequently their ATs, which means that the vehicles need to frequently request new ATs to the PKI.

The IEEE 1609.2 standard [15] defines security data structures, especially secure message and certificate formats, and the processing of those secure messages. Messages' authenticity and integrity is based on digital signatures using ECDSA: several curves with specific parameters and sizes are specified to target different security levels (NIST P-256,

Brainpool-P256 and Brainpool-P384). The confidentiality protection is based on AES symmetric encryption (AES-CCM authenticated encryption). An asymmetric encryption scheme using elliptic curve integrated encryption scheme (ECIES) is provided and is used to transport symmetric encryption keys. In Europe, ETSI has specified a European profile for secure V2X messages and certificates that relies on IEEE 1609.2 and uses the same ECC primitives [16].

Both IEEE Std 1609.2 and ETSI TS 103 097 standards have introduced crypto-agility in the protocol design allowing the choice among various cryptographic primitives. For the deployment of C-ITS services in Europe, the Certificate Policy, published by the European Commission [17], enforces requirements on the supported crypto-algorithm parameters/key lengths for digital signature. It also specifies rules for decision and enforcement by the Certificate Policy Authority (CPA) in case a transition to a higher security level is needed. Even if crypto-agility techniques have been recommended early by the C-ITS Platform, there is still no activities started to address the transition to quantum safe cryptography by the European Commission assisted by the ITS Stakeholders' Expert Group which currently takes the role of CPA.

However, quantum threats on ITS security standards should not be neglected: the cryptographic mechanisms used in ETSI standards are not quantum safe and if a large-scale QC could be built, this would cause a complete loss of trust in the C-ITS trust model. Firstly, regarding the safety messages exchanged between ITS stations, the integrity and data origin authenticity of messages cannot be guaranteed as the signature can be forged. Secondly, as the key management is done by a PKI, also named CCMS (C-ITS Certificate Management System), there are parts of the PKI entities which could be impacted by a quantum computer attack. Many threats could be possible for a quantum attacker, impacting the certificate and trust lists distribution, such as the ECTL (European Certificate Trust List) and CRL (Certificate Revocation List), possibly affecting the whole trusted system entities.

IV. Feasability and performance benchmark

A. Landscape of PQC schemes and benchmarking goals

The research community recently started to work on algorithms resistant to attacks using a QC. The most organised one started in 2017 by NIST which called for submissions of signature and encryption algorithms. The aim of this process is to challenge propositions and at the end standardize the most resistant ones.

Five main families have been identified: code-based, lattice-based, hash-based, isogeny-based and multivariate-based cryptography. In addition, some efforts were done on proposing symmetric key signature schemes, but it did not conclude to a resistant candidate.

The algorithms considered for the benchmarking presented in Section IV.C are the result of the 3rd and 4th round of the NIST competition. The 3rd round (finished in July 2022) points three signatures and one KEM scheme; while the 4th round proposes four more KEM candidates. One of these KEM candidates (i.e. SIKE) is already announced as insecure [26] and as such results about it are not presented in this paper. In this section, we will briefly present the families with at least one candidate in round 3 selection and round 4, and their main characteristics.

Code-based cryptography is based on the theory of error-correcting codes. For some specially constructed codes it is possible to correct many errors, while for random linear codes this is a difficult problem. These cryptosystems are mature (dated from 1978 and the McEliece cryptosystem [20]) and high confidence is expected especially in code-based encryption system. While quite fast, most code-based primitives suffer from having very large key sizes. Some code-based signatures have been designed to offer short signatures at the expense of very large key sizes. However, all code-based signatures submitted to NIST were based on new assumptions and have all been broken. The following KEM candidates advanced in the fourth round: BIKE, Classic McEliece and HQC and their authors are invited to provide improvements on their schemes. No code-based signature algorithm was selected for standardization after the 3rd round of the NIST competition.

It is to be noted that BIKE and HQC are using 'more' special codes to reduce the key size of the public key, which is seen as the main drawback of code-based systems.

Lattice-based cryptography is based on NP-hard problems of high-dimensional lattices, e.g. Shortest Vector Problem (SVP) or Closest Vector Problem (CVP). A lattice is a set of integer linear combinations of a basis vector and the algorithms using this theory consists mainly of linear operations over matrices and polynomials modulo relatively small integers. This kind of operations are highly parallelizable and can be sped up by using vector instructions, multi-thread or multi-core programming. The main drawback of lattice-based cryptography (i.e. large parameter size), is almost resolved as the size of the public key and the ciphertext/signature has been reduced from several gigabytes in the first-generation lattice-based cryptography to several kilobytes in the recent proposals based on the ring LWE (Learning With Errors) or the NTRU (Nth Degree Truncated Polynomial Ring Units) problem. One of the main technical difficulties is that lattice-based KEMs have a small probability of decryption failure, which means that the receiving parties fail to derive a shared secret key for a few ciphertexts. This is the case for the majority of schemes based on lattices (module or ring LWE/Learning With Rounding schemes) or on codes. As this failure is dependent on the secret key, it might leak secret information to an attacker ([22]). Methods to mitigate this issue exist (e.g. [23]) and are quite efficient, but the high overhead induced is not acceptable for practical reasons. As these modifications to the KEM schemes would lead to increased public-key and ciphertext length, the design choice of having imperfect correctness was made for many NIST submissions. One lattice-based KEM (CRYSTALS-Kyber) and two signatures (CRYSTALS-DILITHIUM and Falcon) algorithms have been chosen to be standardized at the end of the 3rd NIST round.

Hash-based cryptography uses cryptographic hash functions. A hash-based signature on one bit is as follows: one picks two random strings; it hashes each of them and publishes the outputs; it then can reveal the first preimage to sign 0 and the second to sign 1. The main disadvantage of this scheme (see

[24]) is that once the secret is revealed, it cannot be used a second time. Based on this scheme, more designs have been developed proposing stateful and stateless versions. A stateful version is useful when the signer needs to keep track of some information (e.g. the number of signatures generated using a given key); while for normal signatures, a stateless signature is enough. Concerning PQC, NIST has already published a document [25] standardizing two stateful hash-based signature schemes: XMSS [21] and LMS [27], while only one stateless hash-based scheme will be finally standardized: SPHINCS+. The main constraint for the stateful schemes is that the key management is crucial: the signer needs to ensure that no individual One-Time Signature (OTS) key is ever used to sign more than one message, because if an attacker were able to obtain digital signatures for two different messages created using the same OTS key, then it would become computationally feasible for that attacker to forge signatures on arbitrary messages. This, in combination to the long key sizes, is sufficient to make them convenient to only a little number of applications (e.g. code signature).

TABLE I. PQC FAMILIES COMPARISON

PQC family	***Function/use***	***To be standardized or 4th round candidate***	***Main characteristics***
Code-based	KEM/encryption, Digital signatures	Classic McEliece, BIKE, HQC	Proposed encryption schemes: high confidence, fast, large public keys Signature schemes: no robust scheme submitted Very low decryption error rate
Lattice-based	KEM/encryption, Digital signatures	Crystals-Kyber, Crystals-Dilithium, Falcon	Encryption: short keys/ciphertext size Signature and public keys size too high (but still the most compact one) Relatively simple and good performance. Very low decryption error rate.
Hash-based	Digital signatures	LMS, XMSS, SPHINCS+	Stateful: crucial key management, long key sizes, suitable for few applications Stateless: easier key management
Isogeny-based	KEM/encryption	SIKE (eliminated from 4th Round)	No robust candidate. Very small key sizes (less than 500 Bytes) but slower performances.

TABLE I. presents a summary of the previous families as well as their main characteristics.

For the selection of future standardized algorithms, NIST identified three evaluation criteria: security, cost/performance, and algorithm/implementation characteristics [5]. Additional criteria like issues relating to patents and maturity of the design may also be considered. To classify the security strength of the submissions, NIST defined the categories seen in TABLE II.

B. Performance needs and constraints of C-ITS solutions

V2X messages size based on ETSI standards is limited by the ITS-G5 technology due to the 5.9 GHz radio band characteristics and the MAC/PHY specification [28].

TABLE II. NIST LEVELS

Level	***Main characteristics***
Level 1	Any attack breaking the relevant security definition must require computational resources comparable to or greater than those required for key search on a block cipher with a 128-bit key.
Level 2	Any attack breaking the relevant security definition must require computational resources comparable to or greater than those required for collision search on a 256-bit hash function.
Level 3	Any attack breaking the relevant security definition must require computational resources comparable to or greater than those required for key search on a block cipher with a 192-bit key.
Level 4	Any attack breaking the relevant security definition must require computational resources comparable to or greater than those required for collision search on a 384-bit hash function.
Level 5	Any attack breaking the relevant security definition must require computational resources comparable to or greater than those required for key search on a block cipher with a 256-bit key.

The maximum transmission unit (MTU) is 1492 bytes at the network layer and only 1428 bytes are available for the payload and the extra security related fields as depicted in Fig.1.

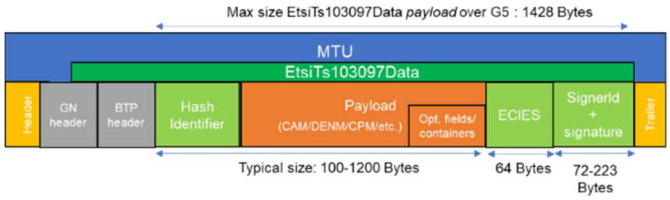

Fig. 1. Logical simplified view of a secured ETSI ITS message

These limits are different from other publications [7] where the study was made to fit US standards; in particular because the max MTU was higher (up to 2304 bytes using DSRC or more for C-V2X) and the fragmentation was considered as a possible option. Signed message broadcasting is not the only use case we should be aware of for the PQC migration of C-ITS standards in Europe. The process and the complete PKI use cases have to be checked also, even if they are less sensitive than the V2X broadcast communications on which we will focus in this study.

Based on standards such as IEEE 1609.2, ETSI TS 103 097 and ETSI EN 302 636-4-1 [29], we made a simplified calculation of the relative sizes to understand what the current situation is. We can see on the TABLE III. that for small messages, cryptographic overhead can represent sometimes more than half of the transmitted data using ECC.

We then listed all the constraints we should satisfy to be compliant with ETSI standards and European profiles (e.g. C2C-CC and C-ROADS) to find suitable solutions that we will be able to implement and demonstrate on TAM OBUs.

Therefore, we took as a baseline the following requirements that the PQC algorithms should satisfy to allow both a minimal impact and backward compatibility in a C-ITS context:

TABLE III. SIMPLIFIED RELATIVE SIZE OF CRYPTOGRAPHY IN ETSI C-ITS MESSAGES

Type (GN over G5)	Size/proportion of cryptography for a 1428 Bytes GN payload	Size/Proportion of cryptography for a 160 Bytes payload	Certificate (cryptography only)	Signature	ECIES parameters (V, C, T)
Signed message (w/ id)	64 Bytes (4.5%)	64 Bytes (28.5%)	0	1 (64 Bytes)	0
Encrypted only message (w/ id and key)	64 Bytes (4.5%)	64 Bytes (28.5%)	0	0	1 (64 Bytes)
Signed & encrypted message (w/ id)	128 Bytes (9.0%)	128 Bytes (44.4%)	0	1 (64 Bytes)	1 (64 Bytes)
Signed message (w/ cert)	160 Bytes (11.2%)	160 Bytes (50%)	1 (96 Bytes)	1 (64 Bytes)	0
Signed & encrypted message (w/ cert)	224 Bytes (15.7%)	224 Bytes (58.3%)	1 (96 Bytes)	1 (64 Bytes)	1 (64 Bytes)

- The fragmentation of messages in multiple packets is currently not used and not supported in applications, so the AT certificate should fit in a single message.
- The AT certificate should be broadcasted every 1 second per vehicle in a basic CAM. AT certificates are explicit certificates following the ETSI profile specified in TS 103 907.
- All the outgoing messages should be signed by the sender: a station with a HSM should be able to sign a message in less than 5 ms to ensure low end-to-end latency.
- All the incoming messages should be verified by the receiver: a station with one dedicated application core for security operations should be able to verify at least 1000 messages per second (802.11p at 12 Mbps/1500 bytes MTU). 2000 messages would be appreciated to have more margins or to support higher bandwidth (24 Mbps).
- Encapsulation and decapsulation of a key should be done within 20 ms each to allow low end-to-end latency.
- The algorithm(s) should be at least at NIST level 1 (see TABLE II.).

C. *Benchmark implementation and analysis*

Five different platforms were chosen to represent the typical OBU of vehicles ranging from 2018 to 2024 and RSU or server like platforms (light edge computing/RSU and light server/PKI). The CPU architecture was the main reason for this selection. Moreover, the chosen hardware is relatively common (at least for OBUs) so the test results can be easily checked. All the requirements defined in section IV. B should be met by the OBU types of this list.

TABLE IV. TAM BENCHMARK PLATFORM LIST

Device	*CPU reference*	*Architecture*	*Station type*
Raspberry Pi 3 B+	BCM2837B0	ARMv8-A Cortex® A53	Low end OBU 2022
Raspberry Pi 4 B	BCM2711	ARMv8-A Cortex® A72	High end OBU 2024
Mini PC	Intel J1900	Intel x86	RSU 2015
Desktop PC	AMD R7 5700G	AMD x86	Server PKI 2021

SUPERCOP [19] v2022.05.06 was used for the benchmark as it provides an open and comprehensive suite of benchmarks. As it provides experimental implementations, improvements are expected in the future regarding the hardware and the software implementations, so the results are a minimum to be expected. We first focus on software implementation with current hardware because we want to know whether it would be possible to migrate the current stations to a new cryptosystem without impact.

The performance figures that are shown below are related to cryptographic operation performance (signature/verification and encapsulation/decapsulation) as well as message and certificate sizes.

TABLE V. PQC SIGNATURE AND/OR VERIFICATION PERFORMANCE RESULTS (LEGEND: RED<1000OP/S; ORANGE>1000OP/S; GREEN >2000OP/S)

	LE OBU 2022		HE OBU 2024		RSU 2015		Server PKI 2021	
Name	**Verification op/sec (147 Bytes)**	**Verification op/sec (1109 Bytes)**	**Verification op/sec (147 Bytes)**	**Verification op/sec (1109 Bytes)**	**Verification op/sec (147 Bytes)**	**Verification op/sec (1109 Bytes)**	**Signature op/sec (1109 Bytes)**	**Verification op/sec (1109 Bytes)**
ECDSA NIST P-256	1 513.0	1 487.1	2 894.6	2 839.1	2 763.0	2 717.1	32 721.7	14 593.4
ECDSA NIST P-384	167.5	167.3	348.3	347.0	352.0	350.6	1 378.5	1 669.0
ECDSA NIST P-521	74.4	74.3	141.4	141.8	362.9	362.0	3 871.8	2 058.0
Dilithium2	1 721.3	1 669.8	4 398.1	4 204.4	3 219.0	3 095.9	2 794.8	12 789.5
Dilithium3	1 104.6	1 082.1	2 760.6	2 686.1	2 022.3	1 973.3	9 185.0	25 577.2
Dilithium5	698.6	689.5	1 710.8	1 677.5	1 244.7	1 225.1	1 468.1	5 221.0
Falcon-512	5 243.6	4 858.4	11 155.2	10 067.5	11 214.7	9 963.2	5 277.9	34 454.1
Falcon-1024	2 529.8	2 441.9	5 361.5	5 100.3	5 559.5	5 248.2	2 618.5	17 807.4
SPHINCS+ 128s-SHAKE256-simple	200.7	198.5	358.0	347.0	246.7	244.2	1.6	1 194.9
SPHINCS+ 192s-SHAKE256-simple	135.9	136.1	241.4	242.0	167.6	167.1	0.8	842.3
SPHINCS+ 256s-SHAKE256-simple	103.1	103.9	184.4	185.4	127.2	127.9	1.2	648.7

TABLE VI. MESSAGE SIZES WITH SIGNATURE IN BYTES (LEGEND: RED>1400BYTES; ORANGE>750 BYTES; GREEN <750BYTES)

Name	Public key length (bytes)	Private key length (bytes)	Sign. length (bytes)	(1) Length of a 160 bytes payload signed message	(2) Length of a 480 bytes payload signed message	(3) Length of a 1120 bytes payload signed message
LMS	64	60	4 756	4 916	5 236	5 876
XMSS	912	19	2 451	2 611	2 931	3 571
Dilithium2	1 312	2 528	2 420	2 580	2 900	3 540
Dilithium3	1 952	4 000	3 293	3 453	3 773	4 413
Dilithium5	2 592	4 864	4 595	4 755	5 075	5 715
Falcon-512	897	7 553	666	826	1 146	1 786
Falcon-1024	1 793	13 953	1 280	1 440	1 760	2 400
SPHINCS+ 128s-SHAKE256-simple	32	64	7 856	8 016	8 336	8 976
SPHINCS+ 192s-SHAKE256-simple	48	96	16 224	16 384	16 704	17 344
SPHINCS+ 256s-SHAKE256-simple	64	128	29 792	29 952	30 272	30 912

Other available results such as key generation time are not shown in this paper. Memory occupancy (RAM and storage use), that can be critical for the HSM, has not been studied. For the comparison tables below, we considered only the algorithms' parameter sets which were selected by NIST at the end of 3rd round. E.g. Kyber provides three parameter sets aiming at different security levels (named Kyber512, Kyber768, Kyber1024).

Certificate profiles and message sizes: To assess whether the cryptographic algorithms fit the size limits for messages and certificate transmission, the resulting size while changing or adding cryptographic fields (signatures, public keys …) has been calculated for multiple profiles. Three message sizes are chosen corresponding to short CAM, larger CAM with optional fields and Collective Perception Message (CPM). We choose to be close to SUPERCOP default values and to take the upper closest value that is a multiple of 16 bytes (because of 128 bits block ciphers). This gives the signed only messages with three payload sizes: 160, 480 and 1120 bytes. As a first approach, addition of certificates in the signed messages have not been considered. Consequently, certificates shall be transmitted in a separate beacon message, e.g. basic CAM, or sent as a response to a Peer-to-Peer certificate distribution request.

As shown in Fig.2, the following types of certificates have been defined: (A) is the reference; a 64 bytes (AT) with only permissions and attributes, (B) is the current certificate we use with ECC, (C) is a replacement of ECDSA with the PQC signature candidate (this serves as a size reference; it does not ensure backwards compatibility and trust), (D) embeds both PQC and ECDSA and (E) is a lighter hybrid approach that is been called "partially PQ hybrid scheme" where we consider the entity to only have ECDSA signature capabilities and assume the ECDSA signature cannot be forged within 1 week [7]. Authorization Authority (AA) certificate has a size of 128 bytes without cryptography (A' format not shown on fig.2). Similarly different types of AA certificates are defined using different signature schemes: ECC signature (B'), PQC only (C'), full hybrid (D') or Partial Hybrid certificate (E').

For encrypted data, we chose, as a first step, a naïve approach with a simple replacement of the ECIES parameters by an AES-256 PQC-encapsulated key (i.e. the ciphertext resulting of the encapsulation of an AES-256 bits key by the PQC KEM candidate).

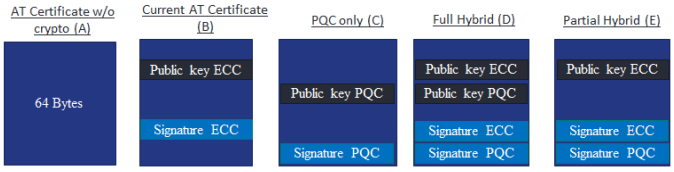

Fig. 2. Certificate types for signature size evaluation

TABLE VII. CERTIFICATES SIZE WITH A SIGNATURE KEY IN BYTES (LEGEND: RED>1400BYTES; ORANGE>750 BYTES; GREEN <750BYTES)

Name	(B) or (C) AT Certificate length	(B') or (C') AA Certificate length	(D) AT Hybrid Certificate length	(D') AA Hybrid Certificate length	(E) AT Sign. Hybrid Certificate length w/ P256	(E') AA Sign. Hybrid Certificate length w/ P256
ECDSA NIST P-256	160	224				
ECDSA NIST P-384	208	272				
ECDSA NIST P-521	262	326				
LMS	4 884	4 948	4 980	5 044	4 916	4 980
XMSS	3 427	3 491	3 523	3 587	2 611	2 675
Dilithium2	3 796	3 860	3 892	3 956	2 580	2 644
Dilithium3	5 309	5 373	5 405	5 469	3 453	3 517
Dilithium5	7 251	7 315	7 347	7 411	4 755	4 819
Falcon-512	1 627	1 691	1 723	1 787	826	890
Falcon-1024	3 137	3 201	3 233	3 297	1 440	1 504
SPHINCS+ 128s-SHAKE256-simple	7 952	8 016	8 048	8 112	8 016	8 080
SPHINCS+ 192s-SHAKE256-simple	16 336	16 400	16 432	16 496	16 384	16 448
SPHINCS+ 256s-SHAKE256-simple	29 920	29 984	30 016	30 080	29 952	30 016

TABLE VIII. MESSAGE SIZE FOR KEM OR ECIES (LEGEND: RED>1400BYTES; ORANGE>750 BYTES; GREEN <750BYTES)

Name	Public key length (bytes)	Private key length (bytes)	Ciphertext of an encapsulated 256 bits key or ECIES parameters for an AES-256 key	(1') Length of a 160 Bytes payload encrypted message	(2') Length of a 480 Bytes payload encrypted message	(3') Length of a 1120 Bytes payload encrypted message
ECIES NIST P-256	32	32	80	240	560	1200
Classic McEliece348864	261 120	6 492	128	288	608	1248
Classic McEliece460896	524 160	13 608	188	348	668	1308
Classic McEliece6688128	104 992	13 932	240	400	720	1360
Classic McEliece6960119	1 047 319	13 948	226	386	706	1346
Classic McEliece8192128	1 357 824	14 120	240	400	720	1360
Kyber512	800	1 632	768	928	1248	1888
Kyber768	1 184	2 400	1088	1248	1568	2208
Kyber1024	1 568	3 168	1568	1728	2048	2688
BIKE L1	1 540	280	1572	1732	2052	2692
BIKE L3	3 082	418	3114	3274	3594	4234
BIKE L5	5 122	580	5154	5314	5634	6274
HQC-128	2 249	40	4481	4641	4961	5601
HQC-192	4 522	40	9026	9186	9506	10146
HQC-256	7 245	40	14469	14629	14949	15589

Signature/Verification "drop in" replacements performance: On the table V, we benchmarked the average verification operations per second for each of the target OBU/RSU platforms and checked whether the defined thresholds can be met. For server platforms, we also measured the average signature operations per second as this is a key performance indicator for the PKI (certificate issuance). Only Crystals-Dilithium and Falcon are meeting the required 2000 op/s (green) threshold. For server/PKI platform, the maximum number of signatures is shown on the left columns. Dilithium is a bit underperforming on ARM based Cortex-A platforms for higher PQC security level but performs better for signature operations. SPHINCS+ being under it, we can eliminate it unless extraordinary software optimizations are discovered (especially for signatures). Additionally, no significant performance difference between smaller and bigger messages has been identified (147 Bytes vs 1109 Bytes).

Signature "drop in" replacements sizes: In addition to the processing performance results, the impact of PQC on signature and public key sizes need to be considered to see whether the corresponding algorithm fits C-ITS constraints. Considering the size limitation factor, only Falcon can be used for signature purpose and through the partial hybrid certificate type we defined (see tables VI and VII). As shown on Table VII, all certificates signed with a PQC signature have too large sizes, except the Partial Hybrid certificates with Falcon signature (orange). For the latest case only, the certificates can be transmitted on a 5.9 GHz safety channel using IEEE 802.11p.

ECIES "drop in" replacements performances: KEM encapsulation and decapsulation times were measured on all our platforms and they were below the specified threshold (20 ms), for all schemes except for BIKE.

ECIES "drop in" replacements sizes: ECIES framework is used in the C-ITS in case there is no AES pre-shared key. The size of transferred ECIES parameters used to rebuild the AES key is 64 bytes (32 bytes ephemeral ECC public key + 16 bytes encrypted key + 16 bytes TAG according to IEEE 1609.2). With an AES-256 bits key, this would reach 80 bytes in total. We summarized the results for the candidates in the table VIII which shows the size of an encrypted message using ECIES or a KEM proposal. Using the same methodology, we can put aside BIKE and HQC as the size of both the ciphertext and the public key is too large. Classic McEliece is also eliminated from the list because of the public key size that cannot be shared within a limited time.

D. Results and discussions

Based on the previous results and to match our requirements, we sum-up the suitability and the limitations found on the list of PQC algorithms which are considered in the NIST competition:

- ECDSA and ECIES are selected (with reserves) and are kept for short period to guarantee required security level in a hybrid way e.g. ECDSA combined with Falcon.
- Dilithium, BIKE, HQC are eliminated: without allowing packet segmentation, it is impossible to fit a signature nor a certificate in a single message.
- Falcon is selected with reserves: Falcon key/signature size is still too big to be efficient, but it is the only PQC signature candidate to meet our performance, relax size and security requirements using a hybrid approach.
- SPHINCS+ is eliminated: its verification performance is too low to be used in C-ITS context. Not to mention that the signatures are also too big and extremely slow.
- McEliece is eliminated: public keys are so large that it would not be possible to share certificate with other stations.
- KYBER is selected with reserves as the key/signature size is still too big to be efficient, but it meets our performance, relax size and security requirements using a hybrid certificate.

V. CONCLUSION AND PERSPECTIVES

There is no algorithm that can be used to simply replace the C-ITS standardized cryptographic algorithms on current hardware without impacting the design of security standards. Best practical solution will certainly imply modifications on the current standard and probably relax some of our assumptions to achieve a correct tradeoff between the radio bandwidth, the performance and the security level of the PQC C-ITS communications. Hence, we decided to focus on the algorithms (Falcon and Kyber) for a transitory period before new and/or more optimized algorithms (and associated hardware accelerators) are available. This could allow to extend the lifetime of current C-ITS stations.

The only approach that seems to have a reasonable impact on current ITS standards and be backward compatible with the legacy ITS station, is the proposed hybrid approach with Falcon and ECDSA. We plan to focus on this solution for our next activities on PQC topic. Concerning encryption, using Kyber seems to be achievable for current use cases and especially for the PKI management protocols, e.g. for encryption of the certificate requests to the PKI but the integration of KEM in the PKI protocol would need more study.

Hopefully, to support the expansion of novel C-ITS applications, new radio technologies are also being developed which offer more bandwidth and larger packets size. We also know that research advance and modification of current algorithms and implementations could lead to major improvements of the current proposals at an equivalent security level. It is also important to mention that measures such as side channel protection may reduce the measured performances. There may be new progresses in cryptanalysis which reduce or call into question the security level of the selected PQC algorithms. Migration work should start now as the process of standardization and certification is anticipated to be very long in C-ITS.

ACKNOWLEDGMENT

This work has been supported by the French government under the "France 2030" program, as part of the SystemX Technological Research Institute.

REFERENCES

[1] M. Mosca and M. Piani, "Quantum Threat Timeline Report 2022," Global Risk Institute in Financial Services(GRI), December 2022.

[2] ENISA Report: Post-Quantum-Cryptography – Current state and quantum mitigation, V2, May 2021.

[3] M. Roetteler, M. Naehrig, K. M. Svore, and K. Lauter: Quantum Resource Estimates for Computing Elliptic Curve Discrete Logarithms, ASIACRYPT 2017, volume 10625 of Lecture Notes in Computer Science, pages 241-270. Springer, 2017.

[4] T. Häner, S. Jaques, M. Naehrig, M. Roetteler, M. Soeken: Improved quantum circuits for elliptic curve discrete logarithms, 27 January 2020, https://arxiv.org/pdf/2001.09580.pdf.

[5] NISTIR.8413: Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process https://doi.org/10.6028/NIST.IR.8413

[6] D. Winkler et al. : Quantum Secure High Performance Automotive Systems, 19th escar Europe, 2021/09/28

[7] N. Bindel et al.: Drive (Quantum) Safe! – Towards Post-Quantum Security for V2V Communications, 23 April 2022, https://eprint.iacr.org/2022/483

[8] NIST: Getting Ready for Post-Quantum Cryptography: Exploring Challenges Associated with Adopting and Using Post-Quantum Cryptographic Algorithms, April 28, 2021.

[9] ANSSI: Views on the Post-Quantum Cryptography transition, January 4, 2022.

[10] BSI: Quantum-safe cryptography – fundamentals, current developments and recommendations, May 2022.

[11] ETSI EN 302 637-2. Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service.

[12] CAR 2 CAR Communication Consortium: Basic System Profile, Release 1.6.3, decemeber 2022, https://www.car-2-car.org/documents/basic-system-profile

[13] C-Roads Platform: Harmonised C-ITS specifications for Europe - Release 2.0, 5 October 2021, https://www.c-roads.eu/platform.html

[14] H. Bakhti, B. Lonc, A. Kaiser : Innovative C-ITS security services implementation and validation by simulation and open-road live trials. ITS European Congress, Nov 2022, Toulouse.

[15] IEEE Std 1609.2™-2016: "IEEE Standard for Wireless Access in Vehicular Environments – Security Services for Applications and Management Messages".

[16] ETSI TS 103 097: "Intelligent Transport Systems (ITS); Security; Security header and certificate formats".

[17] C-ITS Certificate Policy - Release from preparatory phase of C-ITS Delegated Regulation, March 2019. https://cpoc.jrc.ec.europa.eu/index.html

[18] TAM project, 2021. https://www.irt-systemx.fr/en/projets/tam/

[19] D. J. Bernstein, and T. Lange (editors). eBACS: ECRYPT Benchmarking of Cryptographic Systems. https://bench.cr.yp.to (visited in June 2022).

[20] RJ McEliece: A Public-Key Cryptosystem Based On Algebraic Coding Theory, 1978, JPL DSN Progress Report, https://tmo.jpl.nasa.gov/progress_report2/42-44/44N.PDF

[21] IETF RfC 8391: XMSS: eXtended Merkle Signature Scheme, May 2018.

[22] Jan-Pieter D'Anvers, Frederik Vercauteren, and Ingrid Verbauwhede: On the impact of decryption failures on the security of LWE/LWR based schemes, journal IACR Cryptology ePrint Archive, 2018. https://eprint.iacr.org/2018/1089.pdf

[23] Dwork, C., Naor, M., Reingold, O.: Immunizing encryption schemes from decryption errors. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS,vol. 3027, pp. 342–360. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24676-3_21

[24] L. Lamport: Constructing Digital Signatures from a one-way function, 18 October 1979, https://www.microsoft.com/en-us/research/uploads/prod/2016/12/Constructing-Digital-Signatures-from-a-One-Way-Function.pdf

[25] NIST Special Publication (SP) 800-208, Recommendation for Stateful Hash-Based Signature Schemes, October 29, 2020. https://csrc.nist.gov/News/2020/stateful-hash-based-signature-schemes-sp-800-208

[26] W. Castryck, T. Decru: An efficient key recovery attack on SIDH (preliminary version), 2022, https://eprint.iacr.org/2022/975

[27] IETF RfC 8554: Leighton-Micali Hash-Based Signatures, April 2019.

[28] IEEE 802.11-2016 - Standard for Information Technology--Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks--Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[29] ETSI EN 302 636-4-1. GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality.