Requirements for a hybrid TPM based on optimized ML-DSA post-quantum signature

Felipe José Aguiar Rampazzo, Rodrigo de Meneses,
Caio Teixeira, Marco A. Amaral Henriques

`[f233261,r197962]@dac.unicamp.br, [caio,marco]@dca.fee.unicamp.br`

[1] School of Electrical and Computer Engineering
Campinas University (Unicamp), Campinas, SP, Brasil

Abstract. *A Trusted Platform Module (TPM) is used to ensure trust between connected devices by storing device integrity measurements. Both TPM and traditional security systems rely on asymmetric cryptography, which can be vulnerable to quantum computers. This work aims to find the requirements for a TPM secured by traditional and post-quantum algorithms, considering the transition through a hybrid model that remains secure even if the post-quantum algorithm breaks. Then, it analyzes the impact of this hybrid model in a TPM using a hybrid TPM emulated in software.*

1. Introduction

When starting an application or equipment, what guarantees does the user have that the environment has not been altered, either maliciously or otherwise? Ensuring the integrity and confidentiality of an environment is of utmost importance, especially in systems not situated in a fully controlled and secure environment. Integrity of equipment ensures that its data is accurate and has not been tampered with by malicious agents, thus protecting the information content. In this scenario, it is necessary to use a mechanism that certifies the environment remains in a secure state. TPMs are crucial for ensuring the integrity and trustworthiness of computing systems. They verify the authenticity of hardware and software components, using symmetric key schemes and message authentication codes (MACs) within the same device. Additionally, TPMs support the use of public key cryptography algorithms to allow trusted third parties to assess and compare the integrity of different equipment. This process is essential for preventing operational failures, financial losses, service disruptions, and security risks, highlighting the critical role TPMs play in maintaining system integrity and security.

In particular, digital signature algorithms are used to attest to the current state of each evaluated component, and the private key(s) necessary for signing is (are) securely stored in a component specially designed to protect sensitive data: The Trusted Platform Module (TPM), which has only the processing and storage capabilities necessary for executing basic cryptographic functions. As will be detailed in the following sections, it is necessary to modernize current TPMs to be effective against attacks from future quantum computers. The new post-quantum cryptography algorithms recently standardized by NIST require more processing and memory than those currently in use, and their implementation in TPMs presents a challenge to be addressed.

This work is focused on a trusted device (TPM) capable of executing both traditional and new post-quantum algorithms. Preliminary results show that even adopting optimized versions of a post-quantum signature algorithm, the memory capacities of these devices need to be significantly expanded, given the greater computational effort required by the new post-quantum algorithms. Some proposed solutions address memory consumption by optimizing algorithms, but this often results in increased execution time.

2. Preliminary concepts

The TPM is a dedicated hardware module with limited resources that provides security services and a range of cryptographic functions for the equipment to which it is attached. The use of the TPM can be employed in attestation protocols, providing a higher level of security compared to a solution based solely on software. Once data is recorded in the TPM, it cannot be tampered with, ensuring the integrity of the information stored there. This immutability of recorded data allows the TPM to be used as a root of trust — a highly secure component on which all other elements of the equipment rely, as its data cannot be modified without detection. Thus, it is possible to attest to the integrity of the equipment components, ensuring they have not been compromised or replaced. These components can include cryptographic keys or measurements of a software or firmware state. Among the set of cryptographic algorithms available in this version are those based on elliptic curves and RSA. These asymmetric cryptographic algorithms are the foundation of existing remote attestation protocols.

All security mechanisms, including remote attestation mechanisms based on RSA or elliptic curve algorithms, are at risk and can be broken by a future Cryptographically Relevant Quantum Computer (CRQC). The significant increase in projects aiming to build a functional quantum computer in recent years represents a threat to the security of Internet communications. The emergence of a CRQC calls into question the robustness of the public key cryptosystems currently employed in crucial processes such as key exchange and signatures, like those used in TPMs. The security of these cryptographic systems is intrinsically linked to the complexity of solving mathematical challenges, such as factoring large integers into their prime factors (RSA) and computing the discrete logarithm (Diffie-Hellman) over elliptic curves (Elliptic Curve Cryptography – ECC).

The crux of the dilemma lies in the fact that a CRQC has the capability to solve such problems in polynomial time, rendering them obsolete [Shor (1997)]. In the face of this imminent threat, efforts have been undertaken to develop a new generation of cryptographic algorithms designed to operate on conventional systems but withstand potential attacks from these quantum machines. Along this path, the National Institute of Standards and Technology (NIST-USA) initiated the Post-Quantum Cryptography Standardization Process in 2016. The objective was to select new standards of public-key cryptography that demonstrate effective resistance against attacks from CRQCs.

After three rounds of evaluation, NIST chose CRYSTALS-Kyber in the category of public-key encapsulation mechanisms (KEM), and CRYSTALS–Dilithium, Falcon, and $SPHINCS^{+}$, in the digital signatures category [Moody (2022)]. Since standardized algorithms have not yet stood the test of time to definitively prove their security, NIST has decided to extend the search for new proposals in this area. Currently, a fourth round of evaluations is underway to standardize additional algorithms deemed sufficiently robust. The quest for cryptosystems resistant to quantum computer attacks is not limited to NIST. The Internet Engineering Task Force (IETF), through RFC 8391, introduced two post-quantum signature algorithms in 2018 and 2019: the eXtended Merkle Signature Scheme (XMSS) and the Leighton-Micali Hash-Based Signatures (LMS), respectively. Like NIST Sphincs+, XMSS and LMS are hash-based signature algorithms, offering the same security as this signature paradigm but with some operational differences.

Without a CRQC, traditional algorithms remain secure alongside emerging post-quantum cryptography standards. However, during the shift to a post-quantum era, there is concern that attackers may store encrypted data now with plans to decrypt it later using quantum technology. A sudden switch to post-quantum algorithms is not feasible, as these new PQC algorithms, though promising, are still relatively untested and could potentially be compromised by classical computers. The recent example of Rainbow (one of NIST finalists) break illustrates this vulnerability [Beullens (2022)].

To ensure robust security in both current and future post-quantum environments and to guard against threats posed by a future CRQC, a transitional phase is crucial. This shift from traditional cryptography to post-quantum models can be eased through hybrid protocols, which combine traditional and post-quantum algorithms within the same protocol. While this approach aids the transition, implementing hybrid algorithms in a TPM presents significant challenges due to the higher computational demands of PQC algorithms compared to modern asymmetric key algorithms. These challenges include increased processing time and the larger storage required for cryptographic keys, signatures, and encrypted texts. Given the TPM limited processing and storage capabilities, research into optimizing these algorithms and protocols is essential to facilitate an effective transition to the post-quantum era.

3. Related works

As of the writing of this article, no literature or patent databases have been found to integrate the use of TPMs into hybrid protocols combining traditional cryptography (RSA or ECC) and post-quantum cryptography (PQC). The closest to a PQC TPM found was the OPTIGA™TPM SLB 9672 model[1], which implements signatures using the XMSS algorithm, but only to protect its firmware update mechanism. However, this or other PQC algorithms are not made available for use in other scenarios.

Paul et al. brought the functionalities available in traditional TPMs to execute standardized PQC algorithms [Paul et al. (2021)]. The authors' goal is to investigate how current TPM specifications can be utilized in a migration process to a post-quantum world, but exclusively using PQC protocols outside of the TPM, which is used solely for generating random seeds and accelerating hash functions. Kim and Kim use the term ”hybrid TPM” to describe a solution that combines a hardware TPM with a software-based TPM, aiming to overcome the limitations of each model [Kim and Kim (2019)]. While this proposed model could potentially address the computational requirements necessary to support new PQC algorithms, no PQC algorithms were actually implemented.

Gilles et al. employ TPM in an Industrial IoT (IIoT) architecture for equipment authenticity, but their focus is not on PQC algorithms [Gilles et al. (2023)]. On the other hand, Fiolhais et al. survey the limitations of TPM and which characteristics should be

[1] https://www.infineon.com/dgdl/Infineon-Whitepaper_PQC_OPTIGA_TPM_SLB_9673-Whitepaper-v03_00-EN.pdf

altered to better accommodate PQC algorithms [Fiolhais et al. (2020)]. Their analysis is conducted in an emulated software environment and does not consider a hybrid version for a transition period or the additional overhead that hybrid protocols bring.

In Román's works, another technique of secure hardware was used to generate a root of trust [Román and Baturone (2021) and Román et al. (2023)]. They combine the use of Physically Unclonable Functions (PUFs) with Attestation Read-Only Memory (AROM) and hash-based PQC signatures to perform remote attestation of equipment. This application allows a verifier to determine whether an IIoT hardware has been modified, replaced, or had its firmware altered. Although the use of PUFs is cheaper compared to TPM, potential weaknesses of this method need to be further analyzed and understood. Table 1 summarizes the main points of the analyzed articles.

Table 1. Works related to the adoption of PQC in TPM

Author	**PQC**	**PQC algorithms**	**Root of Trust**
(Paul et al., 2021)	✓	Kyber90s and Dilithium	Hardware TPM
(Kim and Kim, 2019)	✗	-	Hardware and Software TPM combined
(Gilles et al., 2023)	✗	-	Hardware TPM
(Fiolhais et al., 2020)	✓	Kyber, NTRU and Dilithium	Software TPM
(Román and Baturone, 2021)	✓	Dilithium and Saturnin	SRAM PUF
(Román et al., 2023)	✓	XMSS and SPHINCS	A-ROM combined with PUF

4. Experiments and results

Emulated TPM Environment: The foundation of the experiment rests on two key software components provided by IBM and Microsoft: the SW-TPM emulator and the TPM Software Stack (TSS), an interface that emulates, via Transmission Control Protocol (TCP), the TPM Command Transmission Interface (TCTI) layer, which is used as an API by applications when requesting services from the TPM. These components were utilized to emulate and test the TPM in a controlled environment, ensuring compliance with ISO/IEC 11889-1:2015 standards. Although other TPM implementations exist (discrete modules, integrated, virtual), the choice of an emulated TPM for conducting experiments is due to the fact that this type of environment is most recommended for prototyping [Trusted Computing Group (2019)].

Given the TPM constraints in terms of computational power, determining the minimal requirements for implementing a PQC-compatible TPM involves careful consideration of several key factors. Based on RFC 7228, which classifies and standardizes constrained nodes, and considering its various aspects to analyze a constrained component, this preliminary work focuses on two critical areas: (i) evaluating the consumption of volatile memory areas and buffers during algorithm execution (peak RAM usage) and (ii) measuring the time required to process operations. To assess peak RAM usage, the Massif tool from the Valgrind framework was employed, with the –stack=yes flag enabled to ensure accurate measurement of total memory consumption. For processing time, Hyperfine was used to measure the response time between a TCTI call to the TPM.

Implementation of hybrid algorithms: The experiment implemented four versions of algorithms designed to resist a quantum attack within the SWTPM [2]. These included two hybrid and two purely post-quantum algorithms. The Module-Lattice-Based Digital Signature Standard (ML-DSA) at security level 2 was chosen as the PQC algorithm base for this analysis. ML-DSA derives from CRYSTALS–Dilithium, a PQC signature algorithm that was selected as one of the finalists in the quest for a quantum-resistant cryptography standard [NIST (2024)]. The results were compared based on the secp256r1 curve (P-256), available in SWTPM and having the same security level as the ML-DSA configuration defined in this work. For constructing the hybrid protocol, we used the Ed25519 elliptic curve for the traditional part of it. Although secp256r1 is used by QSC-OpenSSL for constructing its hybrid protocols, we opted for Ed25519 to explore an equally secure alternative This choice does not compromise the validity of the tests, as Ed25519 is a well-established curve widely studied in the cryptographic community. Future studies may complement this research with analyses using secp256r1 and other "traditional + PQC" combinations for more direct comparisons with QSC-OpenSSL.

Optimizations to reduce memory usage: To identify reduced memory requirements for a TPM with hybrid or pure PQC algorithms, an optimized ML-DSA version was developed based on Bos et al.'s work [Bos et al. (2022)]. In this version, the author reduced memory consumption by generating the matrix A used in the CRYSTALS-Dilithium key calculations on demand: each row of the matrix is created and used within the same memory region. In the reference version of CRYSTALS-Dilithium, and consequently in ML-DSA, this matrix is generated entirely and stored in memory. In addition to this optimization, another was developed for this work. In matrix $\mathbf{A}$, each element consists of a polynomial of degree n. The memory space of each polynomial was also reused, with its content reused in each operation. Although the reduction in memory consumption here is smaller than that of the previous optimization, any decrease in memory usage is crucial, given that TPMs have significant memory constraints for performing operations.

Measurements: To measure the impact that PQC algorithms can exert on TPMs, comparisons were made between the base ECC algorithm (secp256r1) and PQC algorithms, both purely post-quantum and hybrid, at the same security level. The Ed25519 curve offers the same security level as secp256r1, according to NIST[3]. The study aimed to evaluate memory usage in TPMs considering the main operations that will be applied in a future context of remote attestation. Table 2 presents the preliminary results related to peak memory consuption obtained in this work for the different versions of PQC algorithms. The processing time required to execute the Create, Sign, and Verify functions is presented in Figure 1. Versions with the prefix *"H-"* identify hybrid versions (Ed25519 + ML-DSA). The suffix *"_opt"* indicates versions where ML-DSA was optimized. The values *"Delta"* ($\Delta\%$) represent the variation in memory consumption compared to ECC.

Based on data presents in Table 2, we can observe that: (i) There is a significant increase in memory consumption when we compare the ML-DSA versions with ECC, as expected; (ii) Versions with optimized ML-DSA present a notable reduction in memory consumption compared to those using the basic ML-DSA version in all three functions; (iii) The addition of ECC to a protocol based on ML-DSA (hybrid version) has little impact on memory consumption, as it can be seen from the comparison of columns ML-

[2] https://github.com/regras/hybrid-pq-tpm

[3] https://csrc.nist.gov/publications/detail/sp/800-186/final

Table 2. Peak of memory (in KiB) for different operations

Algorithm/ operation	ECC	ML-DSA		ML-DSA_opt		H-ML-DSA		H-ML-DSA_opt	
	KiB	KiB	Δ%	KiB	Δ%	KiB	Δ%	KiB	Δ%
CC_Create	21.7	45.2	108.3	30.1	38.7	45.2	108.3	31.2	43.8
CC_Sign	30.2	58.5	93.7	45.6	51.0	59.6	97.4	45.7	51.3
CC_Verify	31.7	43.2	36.3	28.2	-11.0	44.5	40.4	28.1	-11.4

DSA and H-ML-DSA (with or without optimizations). The reason for this is that in the development of the hybrid method, objects allocated in memory for ECC and no longer used were cleaned up. Therefore, the memory peak was mainly caused by the PQ part of the algorithm.

To gain a better understanding of the effects of hybrid algorithms on a future PQC TPM, a comparison between the processing times of purely PQC versions of ML-DSA and hybrid algorithms, both with and without optimization, can be seen in Figure 1. In all PQC versions, the impact on processing time caused by optimizations is negligible for the Create and Verify functions, and minimal for Sign ($\sim 10\%$ for non-hybrid and $\sim 5\%$ for hybrid versions).

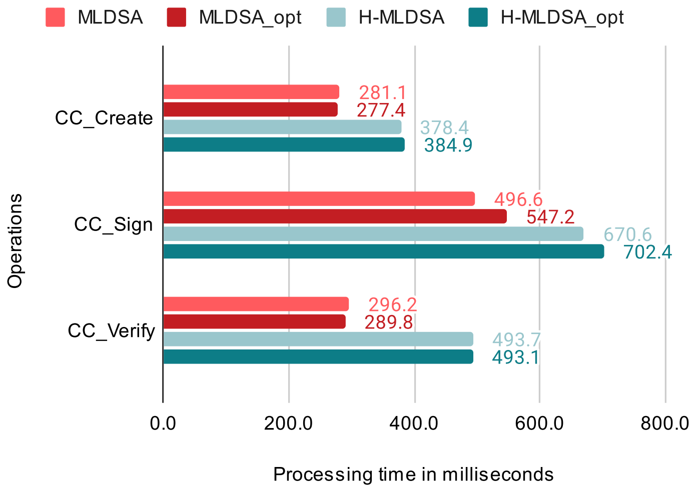

Figure 1. Comparison of processing time (in ms) among PQC versions

5. Conclusions and future works

The memory optimization in hybrid TPM has a significant impact and proves to be a beneficial strategy for both hybrid and non-hybrid TPMs. Although such optimization caused a small increase in processing time for signature verification, it is negligible in the other operations (key creation and signature verification). As future work, this research will evaluate in detail the configuration requirements of a post-quantum TPM with respect to parameters memory (NVRAM) and code memory (ROM), aiming to demonstrate that the implementation of a hybrid TPM (with ML-DSA) is viable for equipment attestation. Efforts will also be made to improve ML-DSA algorithm in order to further minimize its memory and processing requirements.

References

Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997. ISSN 0097-5397. URL https://doi.org/10.1137/S0097539795293172.

Dustin Moody. Status report on the third round of the NIST post-quantum cryptography standardization process. Technical report, 2022. URL https://doi.org/10.6028/nist.ir.8413.

Ward Beullens. Breaking rainbow takes a weekend on a laptop. Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-15978-7. URL https://doi.org/10.1007/978-3-031-15979-4_16.

Sebastian Paul, Felix Schick, and Jan Seedorf. Tpm-based post-quantum cryptography: A case study on quantum-resistant and mutually authenticated tls for iot environments. In *Proceedings of the 16th International Conference on Availability, Reliability and Security*, ARES '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450390514. URL https://doi.org/10.1145/3465481.3465747.

Yongjin Kim and Evan Kim. htpm: Hybrid implementation of trusted platform module. In *Proceedings of the 1st ACM Workshop on Workshop on Cyber-Security Arms Race*, CYSARM'19, page 3–10, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450368407. URL https://doi.org/10.1145/3338511.3357348.

O. Gilles, D. Gracia Pérez, P.-A. Brameret, and V. Lacroix. Securing iiot communications using opc ua pubsub and trusted platform modules. *J. Syst. Archit.*, 134(C), 2023. ISSN 1383-7621. URL https://doi.org/10.1016/j.sysarc.2022.102797.

Luís Fiolhais, Paulo Martins, and Leonel Sousa. Software emulation of quantum resistant trusted platform modules. pages 477–484, 01 2020. doi: 10.5220/0009886004770484.

Roberto Román and Iluminada Baturone. Sealed storage for low-cost iot devices: An approach using sram pufs and post-quantum cryptography. In *Proceedings of the 2021 European Interdisciplinary Cybersecurity Conference*, EICC '21, page 54–59, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450390491. URL https://doi.org/10.1145/3487405.3487651.

Roberto Román, Rosario Arjona, and Iluminada Baturone. A lightweight remote attestation using pufs and hash-based signatures for low-end iot devices. *Future Gener. Comput. Syst.*, 148(C):425–435, 2023. ISSN 0167-739X. URL https://doi.org/10.1016/j.future.2023.06.008.

Trusted Computing Group. Trusted platform module (tpm) 2.0: A brief introduction. Technical report, Trusted Computing Group, 2019.

NIST. *Module-Lattice-Based Digital Signature Standard.* National Institute of Standards and Technology, August 2024. URL https://doi.org/10.6028/NIST.FIPS.204.

Joppe W. Bos, Joost Renes, and Amber Sprenkels. Dilithium for memory constrained devices. In *Progress in Cryptology - AFRICACRYPT 2022*, page 217–235, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-17432-2. URL https://doi.org/10.1007/978-3-031-17433-9_10.